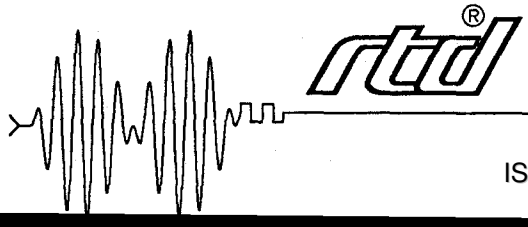
AD510 User's Manual



Real Time Devices, Inc.

"Accessing the Analog World"™

ISO9001 and AS9100 Certified

AD510
User's Manual

REAL TIME DEVICES, INC.
820 North University Drive
Post Office Box 906
State College, Pennsylvania 16804
Phone: (814) 234-8087
FAX: (814) 234-5218

Published by
Real Time Devices, Inc.
820 N. University Dr.
P.O. Box 906
State College, PA 16804

Copyright © 1991 by Real Time Devices, Inc.
All rights reserved

Printed in U.S.A.

Rev. A 9211

TABLE OF CONTENTS

INTRODUCTION *i*-1

Analog-to-Digital Conversion *i*-3
Digital I/O *i*-3
What Comes With Your Board *i*-3
Using This Manual *i*-3
When You Need Help *i*-4

CHAPTER 1 — BOARD SETTINGS 1-1

Factory-Configured Switch and Jumper Settings 1-3
P2 — Base Address (Factory Setting: 200 hex (512 decimal)) 1-4
P3 — Interrupt Channel Selection (Factory Setting: Disabled) 1-4
P5 — External Trigger In (Factory Setting: Disabled) 1-5
S1 — A/D Conversion Rate (Factory Setting: 7.5 Hz) 1-5
Changing the Programmable Gain Settings 1-6

CHAPTER 2 — BOARD INSTALLATION 2-1

Board Installation 2-3
External I/O Connections 2-3
Connecting the Analog Input Pins 2-4
Connecting the Digital I/O Pins 2-4
Connecting the Trigger In and Trigger Out Pins 2-4
Connecting the Status Pin 2-4
Connecting the Reset Drv Pin 2-4

CHAPTER 3 — HARDWARE DESCRIPTION 3-1

A/D Conversion Circuitry 3-3
Analog Inputs 3-3
A/D Converter 3-4
Programmable Peripheral Interface 3-4
Interrupts 3-4

CHAPTER 4 — BOARD OPERATION AND PROGRAMMING 4-1

A/D Conversions 4-3
Initializing the PPI 4-3
Selecting the Channel 4-4
Setting the Gain 4-4
A/D Conversion Modes 4-5
Single Convert Mode 4-5
Continuous Convert Mode 4-7
Cascading Boards 4-7
Digital I/O 4-7
Interrupts 4-7
Defining the I/O Map 4-10
BA + 0: PPI Port A — Read A/D Data (Read Only) 4-10
BA + 1: PPI Port B — Channel Select/Digital Out (Read/Write) 4-11
BA + 2: PPI Port C — A/D Control/Digital I/O (Read/Write) 4-11
BA + 3: PPI Control Word (Read/Write) 4-12
BA + 4: Status (Read Only) 4-13

Example Programs 4-13
BASIC Programs: 4-13
Turbo Pascal Programs: 4-13
Turbo C Programs: 4-13
Lexicon Test and Diagnostics Tool 4-13

CHAPTER 5 — CALIBRATION 5-1

Required Equipment 5-3
A/D Calibration 5-3
Offset Adjustment 5-3
Full-Scale Adjustment 5-5

APPENDIX A — AD510 SPECIFICATIONS A-1

APPENDIX B — CONNECTOR PIN ASSIGNMENTS B-1

APPENDIX C — COMPONENT DATA SHEETS C-1

APPENDIX D — LEXICON 510 ..... D-1

APPENDIX E — WARRANTY E-1

LIST OF ILLUSTRATIONS

1-1	Board Layout Showing Factory-Configured Settings	1-3
1-2	Base Address Jumper, P2	1-4
1-3	Interrupt Channel Selection Jumper, P3	1-5
1-4	External Trigger In, P5	1-5
1-5	A/D Conversion Rate Switch, S1	1-5
1-6	Controlling the Programmable Gains	1-6
2-1	P4 I/O Connector Pin Assignments	2-3
2-2	Analog Input Connection	2-4
2-3	Cascading Two Boards for Simultaneous Sampling	2-5
3-1	AD510 Block Diagram	3-3
4-1	PPI Interconnect Diagram	4-3
4-2	Setting the Gain to 100 Using Bit Set/Reset Function	4-5
4-3	Single Convert Mode Timing Diagram	4-5
4-4	Single Convert Mode Flow Diagram	4-6
4-5	Continuous Convert Mode Timing Diagram	4-7
4-6	Continuous Convert Mode Flow Diagram	4-8
4-7	Cascaded Boards, Single Convert Mode Flow Diagram	4-9
5-1	Adjustable Voltage Source	5-3
5-2	Board Layout Showing Calibration Trimpots	5-4

INTRODUCTION

The AD510 multichannel integrating analog input board turns your IBM PC/XT/AT or compatible computer into a high-performance data acquisition and control system. Installed within a single short or full-size expansion slot in the computer, the AD510 board features:

- Eight single-ended analog input channels,
- 12-bit integrating A/D converter for high stability and exceptional noise immunity,
- Programmable gains of 1, 10, and 100,
- Trigger in and trigger out for cascading boards,
- Nine TTL/CMOS digital I/O lines.

The following paragraphs briefly describe the major functions of the board. More detailed discussions of board functions are included in Chapter 3, *Hardware Description*, and Chapter 4, *Board Operation and Programming*. The board setup is described in Chapter 1, *Board Settings*.

Analog-to-Digital Conversion

The analog-to-digital (A/D) circuitry receives up to eight single-ended analog inputs and converts these inputs into 12-bit digital data words which can then be read and/or transferred to PC memory.

The input voltage range is -5 to +5 volts for a gain of 1 (the range is -0.5 to +0.5 volts for gain = 10; -.05 to +.05 volts for gain = 100). The conversion rate is switch-selectable for either 7.5 or 30 Hz. The 7.5 Hz rate provides high rejection (over 60 dB) of 60 Hz line noise.

A/D conversions are performed by a 12-bit (plus sign bit showing polarity) dual-slope integrating converter. The programmable gains of 1, 10, and 100 let you discern changes in the input voltage as small as 13 microvolts.

The converted data is read and/or transferred to PC memory, one byte at a time, through the PC data bus.

Digital I/O

The AD510 has nine TTL/CMOS-compatible digital I/O lines which can be directly interfaced with external devices or signals to sense switch closures, trigger digital events, or activate solid-state relays. The lines are provided by the on-board 8255 programmable peripheral interface (PPI) chip.

What Comes With Your Board

You receive the following items in your AD510 package:

- AD510 interface board
- Software and diagnostics diskette with example programs in BASIC, Turbo Pascal, and Turbo C; source code
- User's manual

If any item is missing or damaged, please call Real Time Devices' Customer Service Department at (814) 234-8087. If you require service outside the U.S., contact your local distributor.

In addition to the items included in your AD510 package, Real Time Devices offers a full line of board accessories. Key accessories for the AD510 include the MX32 analog input expansion board with 16 single-ended (or 16 differential) input channels, the TB40 terminal board and XB40 prototype/terminal board for prototype development and easy signal access, EX-XT and EX-AT extender boards for simplified testing and debugging of prototype circuitry, and the XC40 single wire flat ribbon cable assembly for external interfacing.

Using This Manual

This manual is intended to help you install your new board and get it running quickly, while also providing enough detail about the board and its functions so that you can enjoy maximum use of its features even in the most complex applications. We assume that you already have an understanding of data acquisition principles and that you can customize the example software or write your own applications programs.

When You Need Help

This manual and the example programs in the software package included with your board provide enough information to properly use all of the board's features. If you have any problems installing or using this board, contact our Technical Support Department, (814) 234-8087, during regular business hours, eastern standard time or eastern daylight time, or send a FAX requesting assistance to (814) 234-5218. When sending a FAX request, please include your company's name and address, your name, your telephone number, and a brief description of the problem.

CHAPTER 1

BOARD SETTINGS

The AD510 board has jumper and switch settings you can change if necessary for your application. The board is factory-configured with the most often used settings. The factory settings are listed and shown on a diagram in the beginning of this chapter. Should you need to change these settings, use these easy-to-follow instructions before you install the board in your computer.

Also note that by installing resistors at two locations on the board, R9 and R10, you can increase the programmable gain settings of x10 and x100 to any values you choose. The procedure for customizing the gain is described at the end of this chapter.

Factory-Configured Switch and Jumper Settings

Table 1-1 lists the factory settings of the user-configurable switch and jumpers on the AD510 board. Figure 1-1 shows the board layout and the locations of the factory-set jumpers. The following paragraphs explain how to change the factory settings.

Table 1-1 — Factory Settings

Switch/ Jumper	Function Controlled	Factory Setting
P2	Sets the base address	300 hex (768 decimal)
P3	Connects the A/D end-of-convert signal to an interrupt channel	Disabled (not connected)
P5	Connects an external trigger for simultaneous triggering of cascaded boards	Disabled (not connected)
S1	Sets the A/D conversion rate	7.5 Hz

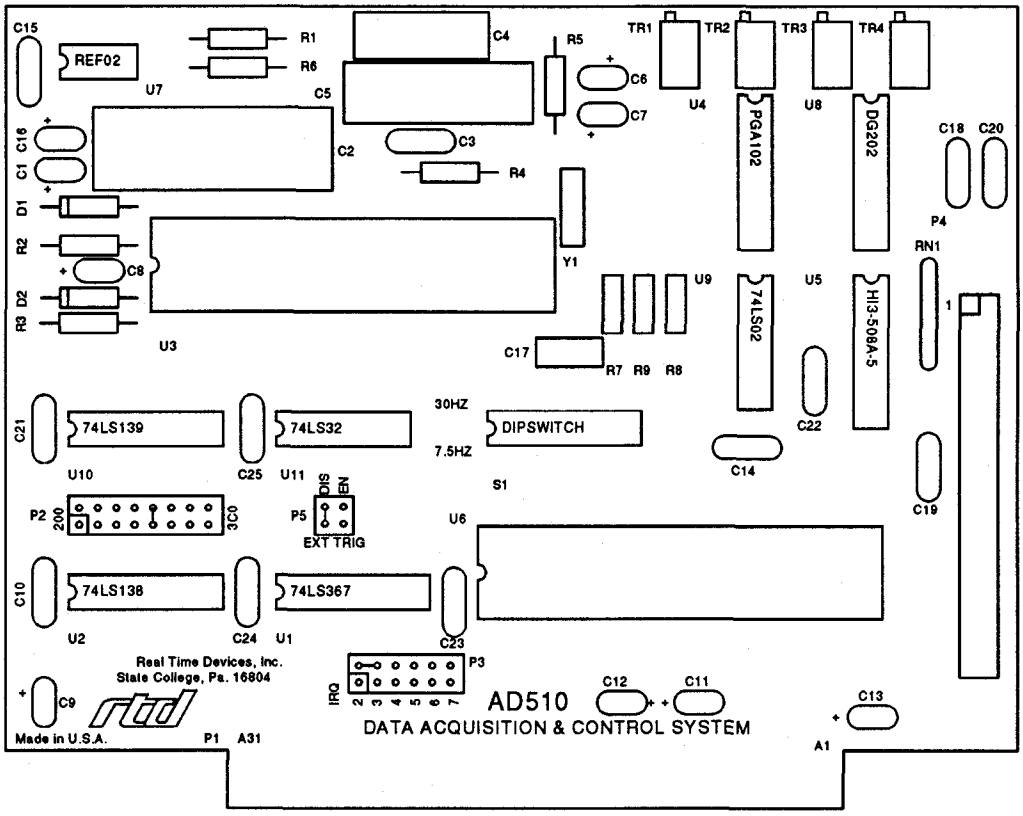


Fig. 1-1 — Board Layout Showing Factory-Configured Settings

P2 — Base Address (Factory Setting: 200 hex (512 decimal))

One of the most common causes of failure when you are first trying your board is address contention. Some of your computer's I/O space is already occupied by internal I/O and other peripherals. When the AD510 board attempts to use I/O address locations already used by another device, contention results and the board does not work.

To avoid this problem, the AD510 has a header connector, P2, which lets you select any one of eight starting addresses in the computer's I/O. Should the factory setting of 300 hex (768 decimal) be unsuitable for your system, you can select a different base address. These addresses are, from left to right on P2:

Hexadecimal	Decimal
200	512
240	576
280	640
2C0	704
300	768
340	832
380	896
3C0	960

To change the base address setting, remove the jumper from the factory setting (300 hex) and, using Figure 1-2 as a guide, install it in the desired location. Record the new base address setting on the table inside the back cover of this manual.

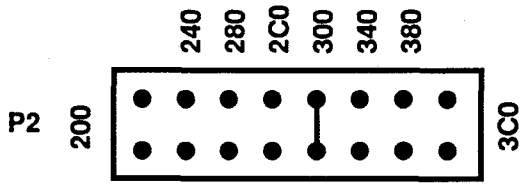


Fig. 1-2 — Base Address Jumper, P2

P3 — Interrupt Channel Selection (Factory Setting: Disabled)

Header connector P3, located near the bottom center of the board, lets you connect the A/D converter's end-of-convert signal to any of the computer's interrupt channels, IRQ2 (highest priority channel) through IRQ7 (lowest priority channel). The jumper is stored across the top two leftmost pins, as shown in Figure 1-3. By placing this jumper vertically across the pins of one of the IRQ channels, the end-of-convert signal can be used to generate interrupts.

Before trying to use interrupts, you must be familiar with the procedure for initializing the interrupt vectors and the PC's interrupt controller, and setting up the interrupt handling routines. These procedures are beyond the scope of this manual, but must be understood to effectively use interrupts in your computer system.

Also, be careful to avoid contention with other devices that may use interrupts in your computer when you choose your interrupt channel. To avoid contention, use the table inside the back cover of this manual to record the interrupt channel you have selected.

It is also very important to note that the board interrupt source is a TTL totem-pole (push/pull) type output; it is **not** open-collector. Therefore, **do not** connect this interrupt to any other interrupt output!

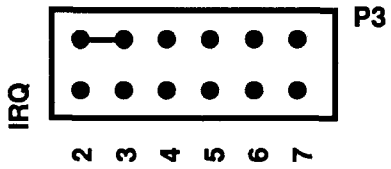


Fig. 1-3 — Interrupt Channel Selection Jumper, P3

P5 — External Trigger In (Factory Setting: Disabled)

P5, shown in Figure 1-4, enables and disables the external trigger input. When the jumper is set to the enabled position, the external trigger in pin (pin 10) at I/O connector P4 is connected to the A/D converter so that two or more boards can be run synchronously in a "master/slave" configuration. Note that this header connector enables and disables the trigger in only; it does not affect the trigger out.

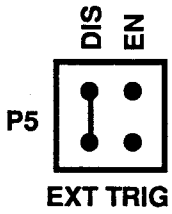


Fig. 1-4 — External Trigger In, P5

S1 — A/D Conversion Rate (Factory Setting: 7.5 Hz)

DIP switch S1, shown in Figure 1-5, configures the board to perform A/D conversions at a rate of 7.5 conversions per second (7.5 Hz) or 30 conversions per second (30 Hz). The four switches operate as a group. When all of the switches are in the DOWN (closed) position, the conversion rate is 7.5 Hz. This setting provides maximum rejection of 60 Hz line noise. When all of the switches are in the UP (open) position, the conversion rate is 30 Hz. When changing the settings, make sure ALL FOUR switches are set to the same position.

Note that the board has been factory-calibrated for a 7.5 Hz rate. If you change the rate, you may need to recalibrate the board. Chapter 5, *Calibration*, explains the procedures.

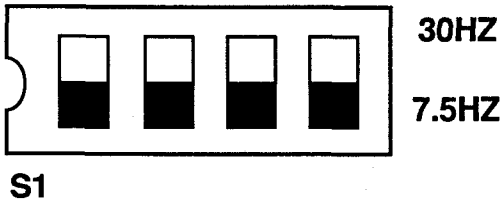
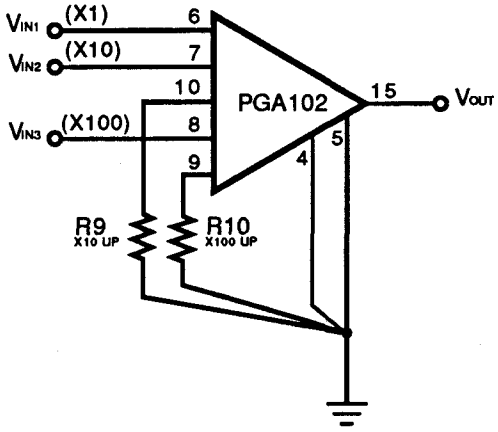


Fig. 1-5 — A/D Conversion Rate Switch, S1

Changing the Programmable Gain Settings

The AD510 board has programmable gains of 1, 10, and 100. For even greater gain flexibility, two empty resistor locations are provided on your board so that you can select the gain values you want. You can set the x10 gain input to any value **greater than** x10, and the x100 gain input to any value **greater than** 100. Note that the gains can only be increased from their base values; they cannot be decreased. Resistor R9 controls the x10 input and should be used to cover gains from 10 to 99. Resistor R10 controls the x100 input and should be used to set gains of greater than 100. Figure 1-6 shows how these resistors are connected to the PGA. The formulas will help you find the correct resistance value for your desired gain. Note that the further you get from the PGA's set gains of 10 and 100, the more gain drift you will have, which can cause errors in your readings.



R9 = [10.8KΩ/(Desired Gain - 10)]

R10 = [108kΩ/(Desired Gain - 100]

Gain/Resistance Values			
Gain	**R9 Value**	**Gain**	**R10 Value**
20	1.08KΩ	200	1.08KΩ
30	540Ω	300	540Ω
40	360Ω	400	360Ω
50	270Ω	500	270Ω

Fig. 1-6 — Controlling the Programmable Gains

CHAPTER 2

BOARD INSTALLATION

The AD510 board is easy to install in your IBM PC/XT/AT or compatible computer. It can be placed in any slot, short or full-size. This chapter tells you step-by-step how to install and connect the board.

Board Installation

Keep the board in its antistatic bag until you are ready to install it in your computer. When removing it from the bag, hold the board at the edges and do not touch the components or connectors.

Before installing the board in your computer, check the jumper settings. Chapter 1 reviews the factory settings and how to change them. If you need to change any settings, refer to the appropriate instructions in Chapter 1. Note that incompatible jumper settings can result in unpredictable board operation and erratic response.

To install the board:

1. Turn OFF the power to your computer.
2. Remove the top cover of the computer housing (refer to your owner's manual if you do not already know how to do this).
4. Select any unused short or full-size expansion slot and remove the slot bracket.
5. Touch the metal housing of the computer to discharge any static buildup and then remove the board from its antistatic bag.
6. Holding the board by its edges, orient it so that its card edge (bus) connector lines up with the expansion slot connector in the bottom of the selected expansion slot.
7. After carefully positioning the board in the expansion slot so that the card edge connector is resting on the computer's bus connector, gently and evenly press down on the board until it is secured in the slot.

   NOTE: Do not force the board into the slot. If the board does not slide into place, remove it and try again. Wiggling the board or exerting too much pressure can result in damage to the board or to the computer.
8. After the board is installed, secure the slot bracket back into place and put the cover back on your computer. The board is now ready to be connected via the external I/O connector at the rear panel of your computer.

External I/O Connections

Figure 2-1 shows the AD510's P4 I/O connector pinout. Refer to this diagram as you make your I/O connections.

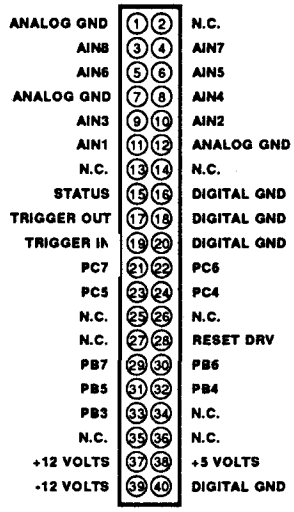

ANALOG GND 1 2 N.C.
AIN8 3 4 AIN7
AIN6 5 6 AIN5
ANALOG GND 7 8 AIN4
AIN3 9 10 AIN2
AIN1 11 12 ANALOG GND
N.C. 13 14 N.C.
STATUS 15 16 DIGITAL GND
TRIGGER OUT 17 18 DIGITAL GND
TRIGGER IN 19 20 DIGITAL GND
PC7 21 22 PC6
PC5 23 24 PC4
N.C. 25 26 N.C.
N.C. 27 28 RESET DRV
PB7 29 30 PB6
PB5 31 32 PB4
PB3 33 34 N.C.
N.C. 35 36 N.C.
+12 VOLTS 37 38 +5 VOLTS
-12 VOLTS 39 40 DIGITAL GND

Fig. 2-1 — P4 I/O Connector Pin Assignments

Connecting the Analog Input Pins

Connect the high side of each analog input to one of the analog input channels, AIN1 through AIN8, and connect the low side to any one of the three ANALOG GND signals (P4-1, 7, or 12). Figure 2-2 shows how these connections are made.

NOTE: It is good practice to connect all unused channels to ANALOG GND, as shown with channel 8 in the diagram below. Failure to do so may affect the accuracy of your conversion results.

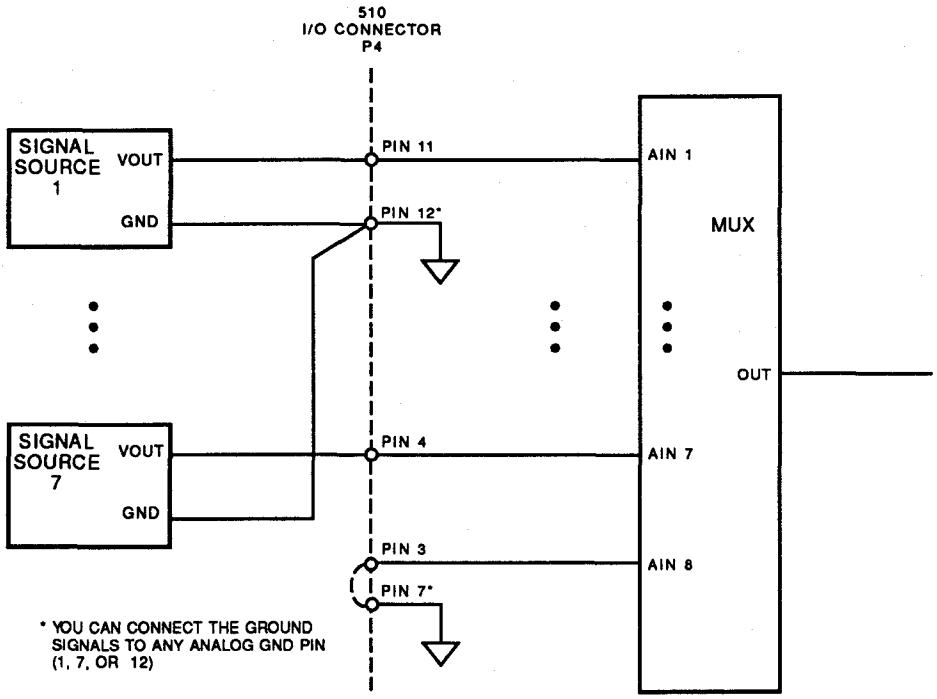


Fig. 2-2 — Analog Input Connection

Connecting the Digital I/O Pins

For all digital I/O connections, PB3-7 and PC4-7, the high side of an external signal source or destination device is connected to the appropriate signal pin on the I/O connector, and the low side is connected to any DIGITAL GND.

Connecting the Trigger In and Trigger Out Pins

The AD510 board has an external trigger input (P4-19) and output (P4-17) so that two or more boards can be cascaded and run synchronously in a "master/slave" configuration. By cascading two (or more) boards as shown in Figure 2-3, they can be triggered to start an A/D conversion at exactly the same time.

Connecting the Status Pin

The STATUS output can be connected to an external circuit which monitors the status of the A/D converter.

Connecting the Reset Drv Pin

The RESET DRV pin can be used to connect the RESET signal generated by the PC to external circuitry.

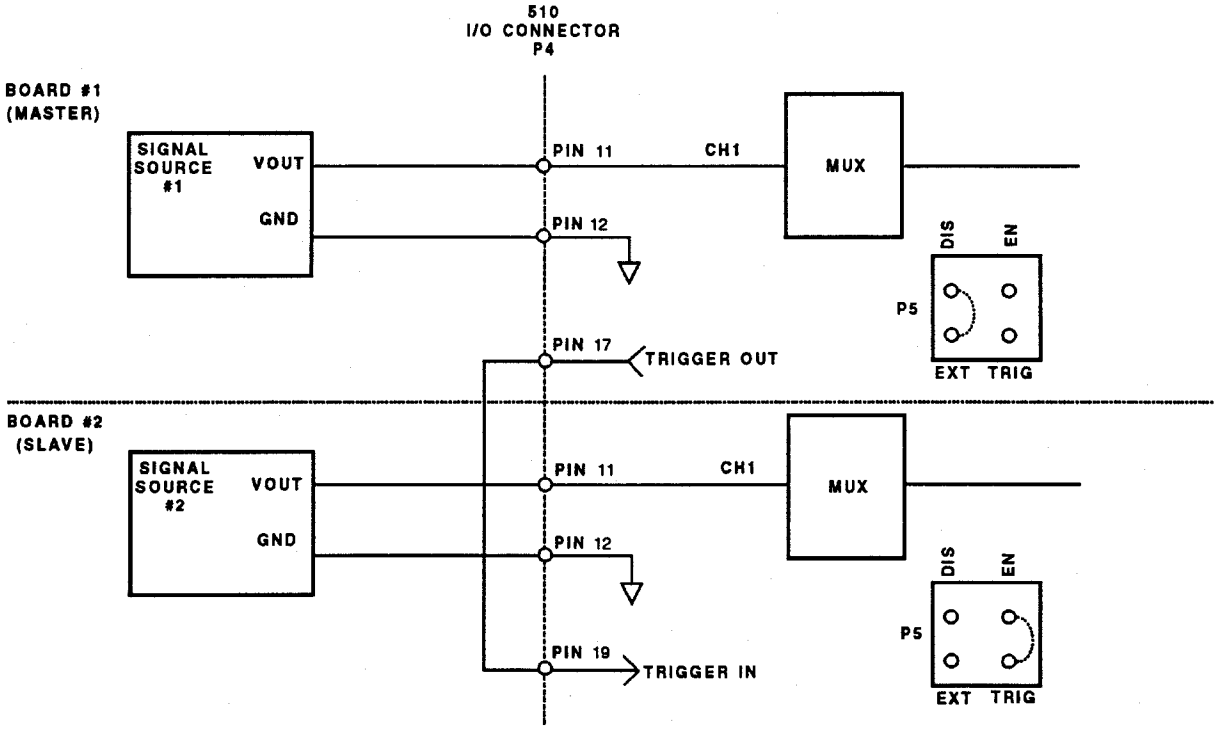


Fig. 2-3 — Cascading Two Boards for Simultaneous Sampling

CHAPTER 3

HARDWARE DESCRIPTION

This chapter describes the features of the AD510 hardware. The major circuits are the A/D and the programmable peripheral interface which includes the digital I/O lines. Board interrupts are also described in this chapter.

The AD510 board has two major circuits, the A/D and the programmable peripheral interface (PPI) which includes the digital I/O lines. Figure 3-1 shows the block diagram of the board. This chapter describes hardware which makes up the major circuits. It also discusses interrupts.

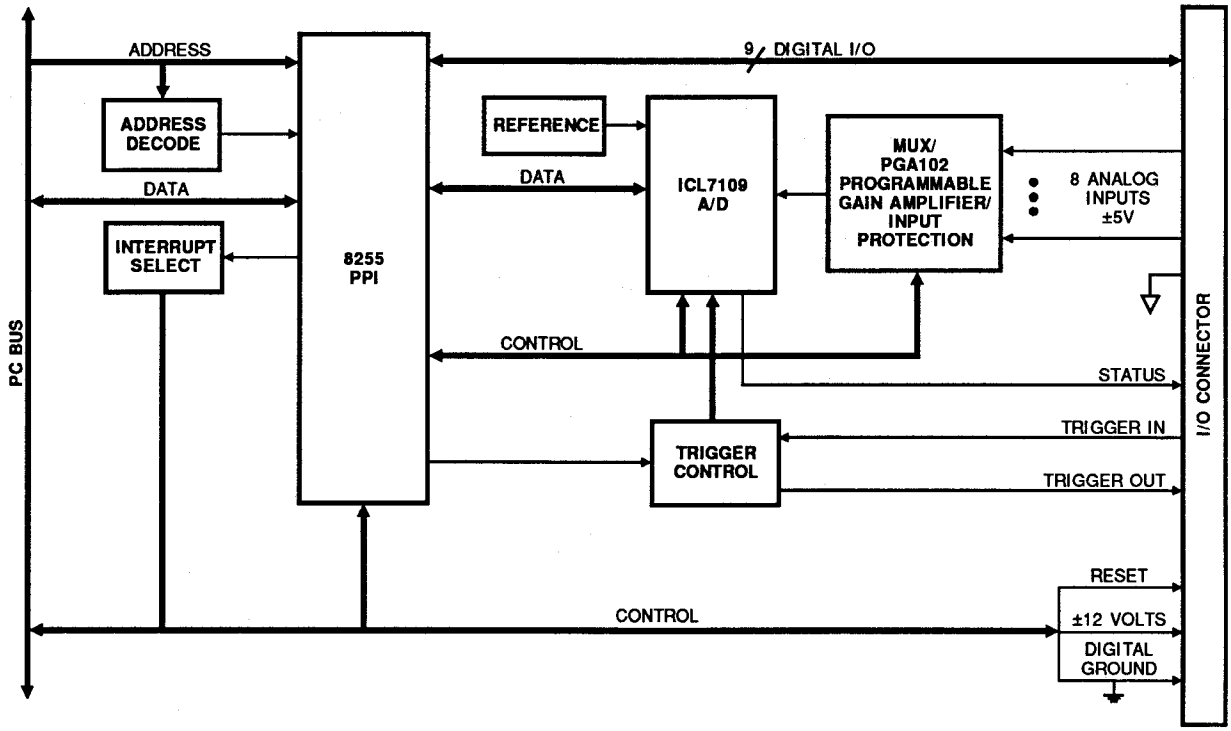


Fig. 3-1 — AD510 Block Diagram

A/D Conversion Circuitry

The AD510 board performs analog-to-digital conversions on up to eight analog input channels. The following paragraphs describe the A/D circuitry.

Analog Inputs

The analog input range is -5 to +5 volts for a gain of 1. Gain is used to match the input voltage levels being measured as closely as possible with the board's input voltage range. For analog inputs ranging between ±500 millivolts, a gain of 10 can be used; and for analog inputs ranging between ±50 millivolts, a gain of 100 can be used. In addition, the x10 and x100 gains can be increased to any desired value as described at the end of Chapter 1. Because it reduces the input voltage range, increasing the gain also increases the resolution of the conversion. For example, when the gain is set to 100, voltage changes of 13 microvolts are reflected in the digitized data. Although overvoltage protection to ±35 volts is provided at the input, you must be careful, especially when using the gains, not to connect excessive input voltages to the board.

Eight single-ended analog input channels are available on the AD510 board. The channels are connected to the A/D converter through a Harris 508A multiplexer. The active channel is selected through software, as described in Chapter 4.

A/D Converter

The dual slope integrating A/D converter performs conversions at a rate of either 7.5 or 30 times per second, depending on the setting of DIP switch S1. The output is a 12-bit data word plus a sign bit which tells you whether the digital value represents a positive or a negative voltage.

The converter samples the input voltage over a window of time before performing the conversion. The converter measures the time required to charge and discharge a capacitor. A counter and a reference in the A/D chip determine the integration period. Because the input signal is sampled for a specified period instead of being instantaneously captured by a sample-and-hold device, spikes and glitches which may be present at the input are averaged out. The result is a highly accurate conversion.

Programmable Peripheral Interface

The programmable peripheral interface (PPI) is used for digital I/O functions. This high-performance TTL/CMOS compatible chip has 24 digital I/O lines divided into two groups of 12 lines each:

Group A — port A (8 lines) and port C upper (4 lines);
Group B — port B (8 lines) and port C lower (4 lines).

Fifteen lines are used for on-board functions. Port A's eight lines are used to read the 8-bit MSB and LSB outputs from the A/D converter. Port B's lower three lines are used to control the channel selection. Port C's lower four lines are used to control the gain selection and the A/D converter. The nine remaining lines in the groups, port B, bits 3 through 8, and port C upper, are available for your use. Chapter 4 explains how you can control these lines.

Interrupts

The AD510 board provides an end-of-convert interrupt signal which can be used to interrupt the computer when an A/D conversion is completed. This signal is the inverse of the status signal generated by the A/D converter and can be connected to any one of interrupt lines IRQ2 through IRQ7 by an on-board jumper installed on header connector P3. The board is shipped with the interrupt signal disabled. If you connect the end-of-convert to one of the interrupt channels, an interrupt will occur when the line transitions from low (converting) to high (not converting). We recommend that you have an understanding of how to use interrupts in your system before you connect the end-of-convert to an IRQ channel.

CHAPTER 4

BOARD OPERATION AND PROGRAMMING

This chapter describes the operating modes and provides flow diagrams and a complete description of the I/O map to aid you in programming your AD510 board. The example programs included on the disk in your board package are listed at the end of this chapter. These programs, written in BASIC, Turbo Pascal, and Turbo C, include source code to simplify your applications programming.

A/D Conversions

Before you are ready to start taking A/D conversions, you must initialize the programmable peripheral interface (PPI), set the gain, and select the channel (when you initialize the PPI, the gain and channel are automatically set to 1). The software provided with your board contains example programs for board initialization.

You can monitor the conversion status using the status bit at I/O address location BA + 4. When bit 0 of this word goes high, it means a conversion is in progress. When it goes low, the conversion is completed. When you are monitoring the status line, make sure you see it go high and then low before you assume the conversion is completed. This line does not go high until slightly after the Start Convert line starts a conversion, which means that you could read a false low when the conversion first begins. (Remember that the status signal is inverted to derive the end-of-convert signal which can be monitored through an IRQ line. The end-of-convert line is low during a conversion and goes high when the conversion is completed.)

Initializing the PPI

Several of the lines of the PPI control the A/D converter, select the channel, and set the programmable gain. The 15 lines which are used during A/D conversion are the eight lines of port A (PA0 through PA7), the three lower lines of port B (PB0 through PB2), and the four lower lines of port C (PC0 through PC3). Port A's eight lines are used to carry the converted data. This data is output from the A/D converter in two 8-bit words, an MSB and an LSB. The three lower lines of port B are used to select the channel. The four lines of port C lower are used to control the gain selection and A/D conversion. Figure 4-1 shows how the PPI interfaces with the converter, multiplexer, and programmable gain amplifier.

To set these lines up so that you can control the gain and channel select and take A/D conversions, the PPI must be initialized whenever you power up or reset your system. This is done by writing data to the PPI control word at I/O address location BA + 3. The I/O map is defined later in this chapter. The PPI must be set up like this:

1	0	0	1	X	0	0	0
D7	D6	D5	D4	D3	D2	D1	D0

X = user defined (see BA + 3 register definition later in this chapter)

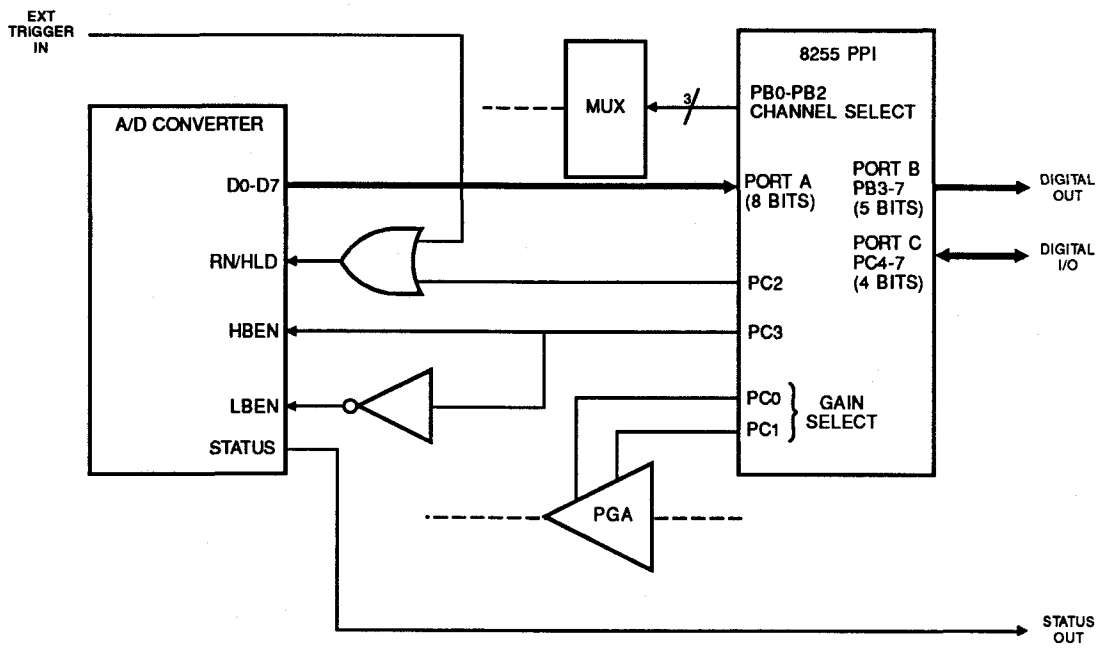


Fig. 4-1 — PPI Interconnect Diagram

Selecting the Channel

You can select the active channel by writing to lines PB0, PB1, and PB2 of port B, I/O address location BA + 1. When you initialize the PPI, the channel select is automatically set to 1. The channel select settings are:

Channel	PB2	PB1	PB0
AIN1	0	0	0
AIN2	0	0	1
AIN3	0	1	0
AIN4	0	1	1
AIN5	1	0	0
AIN6	1	0	1
AIN7	1	1	0
AIN8	1	1	1

Note that when you write to port B to change the channel selection, you must preserve the other five bits of data (PB3 through PB7) as you had them if you want those settings to be unchanged. You can do this by first reading the port B data and then ANDing its contents with hex F8 (decimal 248). This resets PB0-2 while preserving PB3-8. Then, OR this result with the desired channel (PB3-8 must be 0 in your channel select word) and write the result back to port B (BA + 1). The example programs show you how to do this.

Setting the Gain

You can set the gain by writing to lines PC0 and PC1 of port C, I/O address location BA + 2. When you initialize the PPI, the gain is automatically set to 1. The gain control settings are:

Gain	PC1	PC0
x1	0	0
x10	1	0
x100	0	1
Invalid	1	1

Note that when you write to port C to change the gain setting, you must preserve the other six bits of data (PC2 through PC7) as you had them if you want those settings to be unchanged.

There is a way to individually set and reset the lines of port C by writing the correct data to the control word at address location BA + 3. For example, you can individually set (to logic 1) or reset (to logic 0) PC0 and PC1 by writing two sets of data to the **control word** (not to port C!) as shown in Figure 4-2. This example shows how to set the gain to 100. When the most significant bit, D7, of the control word is set to 0, it activates the bit set/reset function which lets you individually set or reset any one of port C's bits:

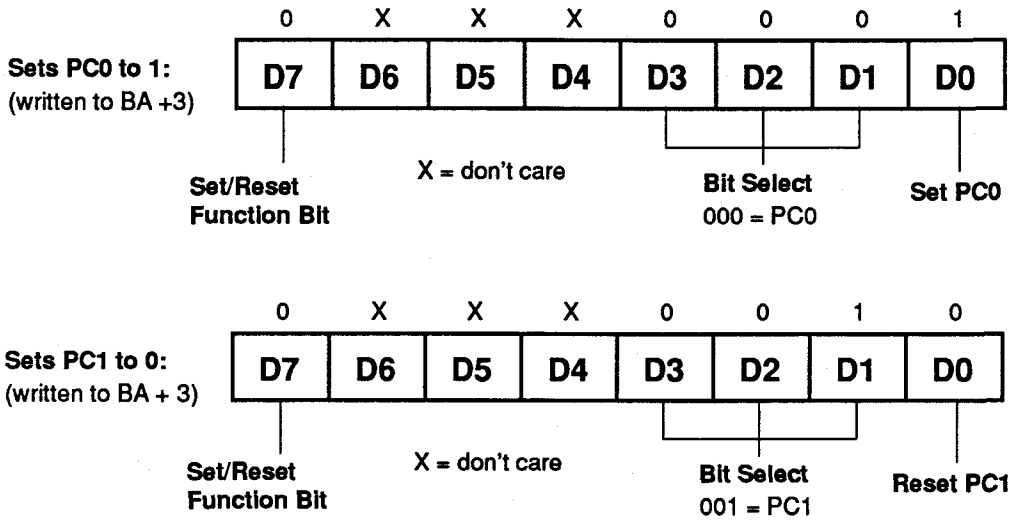

Sets PC0 to 1:
(written to BA +3)
0 X X X 0 0 0 1
D7 D6 D5 D4 D3 D2 D1 D0
Set/Reset Function Bit
X = don't care
Bit Select
000 = PC0
Set PC0
Sets PC1 to 0:
(written to BA + 3)
0 X X X 0 0 1 0
D7 D6 D5 D4 D3 D2 D1 D0
Set/Reset Function Bit
X = don't care
Bit Select
001 = PC1
Reset PC1

Fig. 4-2 — Setting the Gain to 100 Using Bit Set/Reset Function

A/D Conversion Modes

The A/D converter can perform conversions in two modes, *Single Convert* and *Continuous Convert*. The board is typically used in the *Continuous Convert* mode.

Two lines from PPI port C, PC2 and PC3, are used to control the conversions. These lines can be set by writing to PPI port C (BA + 2) or by using the single bit set/reset function described in the previous section, *Setting the Gain*, and under BA + 3 in the *Defining the I/O Map* section later in this chapter. PC2 controls the Start Convert function. Whenever PC2 is high, conversions can be started. PC3 determines which byte of the converted data, the MSB or the LSB, is to be read at PPI port A. When PC3 is low, the MSB is read; when it is high, the LSB is read. For each conversion, the MSB is typically read first, followed by the LSB.

- **Single Convert Mode**

The *Single Convert* mode lets you perform a single A/D conversion each time you pulse the Start Convert line, PC2, high. Figure 4-3 shows the timing diagram for this mode, and Figure 4-4 provides a flow diagram.

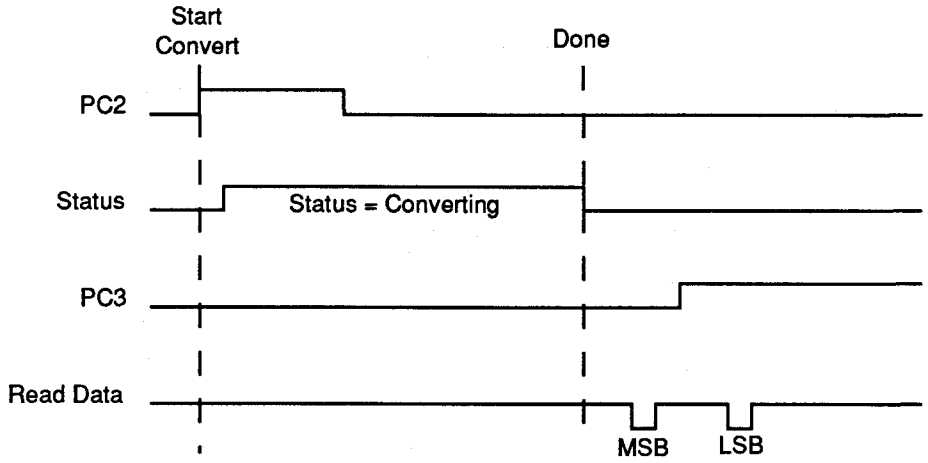


Fig. 4-3 — Single Convert Mode Timing Diagram

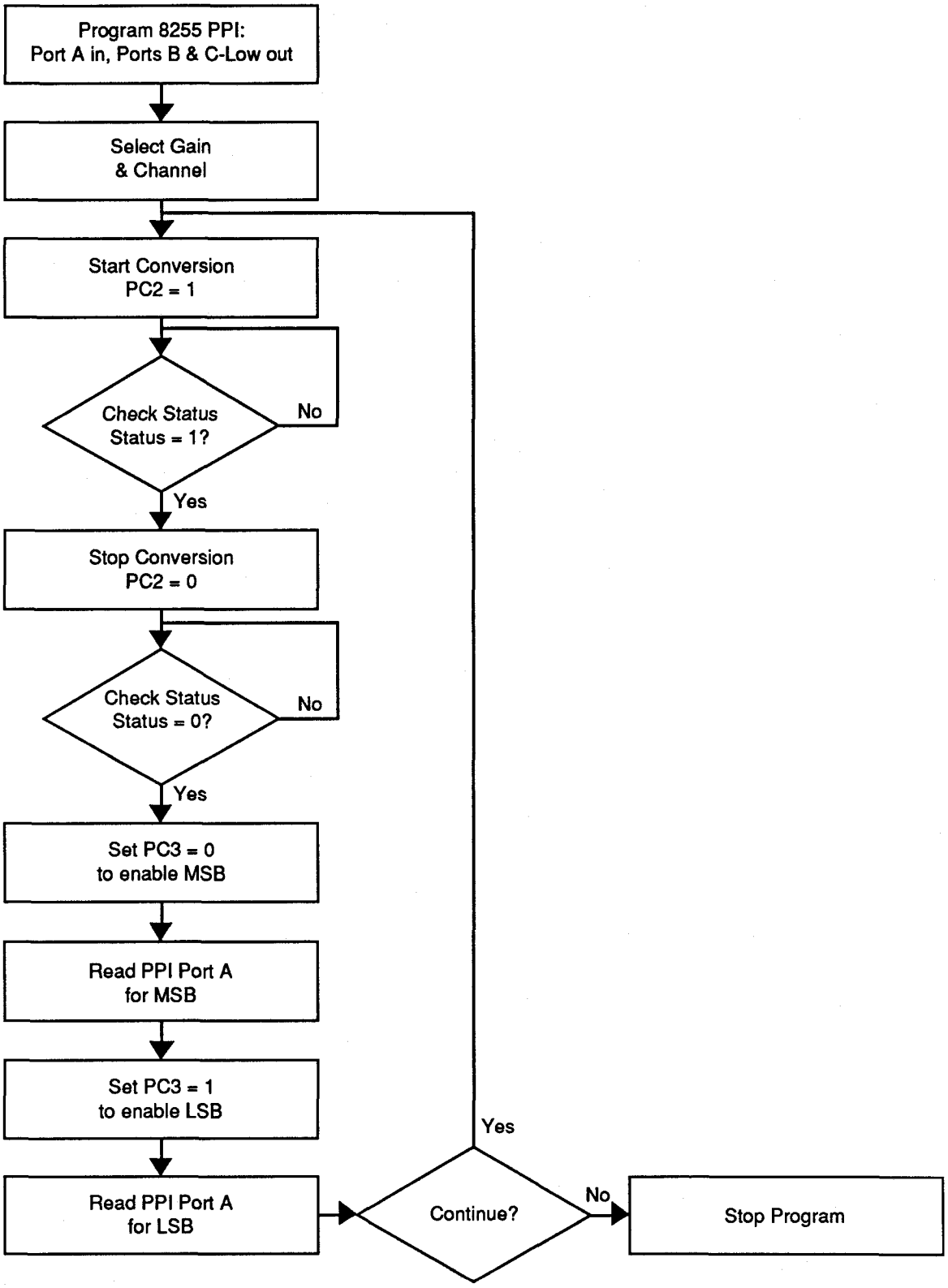


Fig. 4-4 — Single Convert Mode Flow Diagram

- **Continuous Convert Mode**

The *Continuous Convert* mode lets you perform continuous A/D conversions by keeping the Start Convert line, PC2, high. Figure 4-5 shows the timing diagram for this mode, and Figure 4-6 provides a flow diagram.

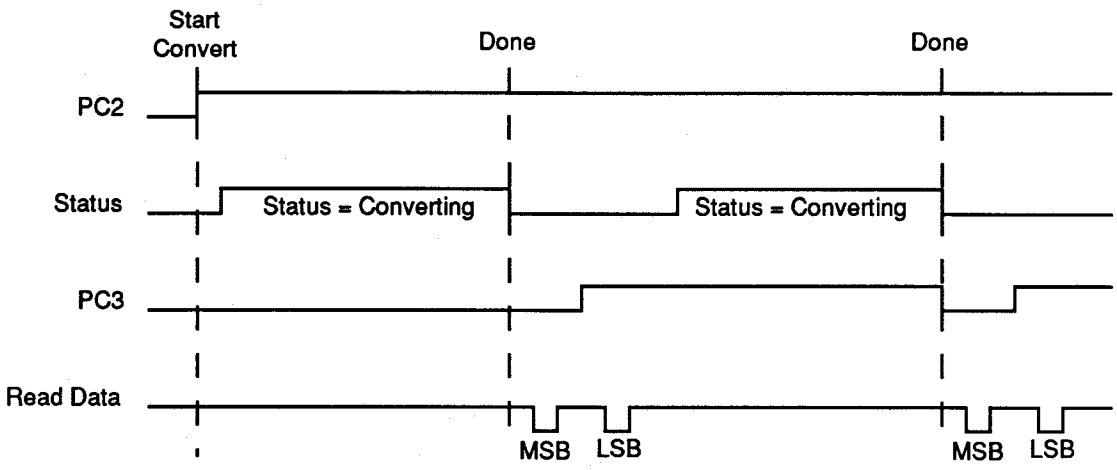


Fig. 4-5 — Continuous Convert Mode Timing Diagram

- **Cascading Boards**

Two or more boards can be cascaded and triggered so that A/D conversions are performed simultaneously on each board. Figure 4-7 provides a flow diagram for cascaded operation. Chapter 2 shows how to connect the boards for simultaneous triggering.

Digital I/O

The programmable peripheral interface has 24 digital I/O lines, nine of which are available for your use. The 24 lines are divided into three 8-bit ports, port A (PA0-7), port B (PB0-7), and port C. Port C is further subdivided into port C lower (PC0-3) and port C upper (PC4-7). Port A, the three lowest bits of port B, and port C lower are already used by the board for A/D conversion, channel selection, and gain control. The rest of the port B bits (PB3 through PB8) and port C upper can be used in this PPI operating mode only:

Mode 0 — Basic input/output. Lets you use simple input and output operations for a port. Data is written to or read from the specified port. Port B can be used as an output port only.

NOTE: If you know about the operation of the 8255 PPI or read the included data sheet, then you may also know that a two other modes, mode 1 (strobed input/output) and mode 2 (strobed bidirectional input/output), exist for the chip. However, since these modes require the use of port C lower or port A, they are not available on the AD510 board.

The PPI is configured by writing a control word to its internal register at I/O address location BA + 3. Seven of the eight bits in the control word are already defined by board initialization requirements. The remaining bit, bit 3, configures port C upper as an input or output port. See the section on *Defining the I/O Map* later in this chapter for a complete definition of the PPI control word and ports A, B, and C data words. See the 8255 PPI data sheet in Appendix C for more operational information.

Interrupts

The A/D converter status signal is inverted and sent to header connector P3 as an end-of-convert signal where it can be jumpered to one of six interrupt channels, IRQ2 through IRQ7. The end-of-convert signal is low whenever an A/D conversion is in progress. When the signal transitions to a high (logic 1) state, an interrupt is sent to the computer telling it that the conversion is completed. Your board is shipped with the interrupt disabled.

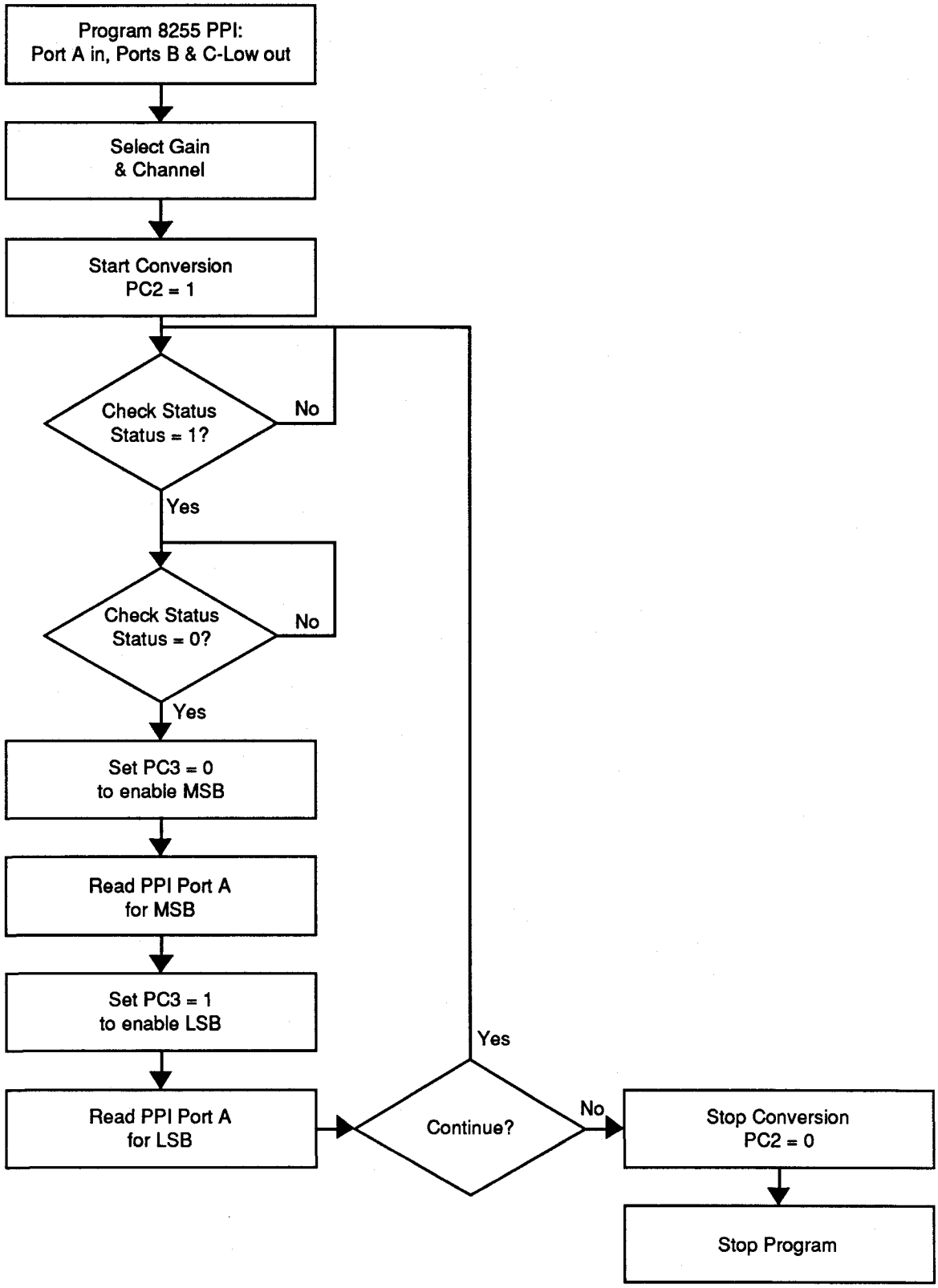


Fig. 4-6 — Continuous Convert Mode Flow Diagram

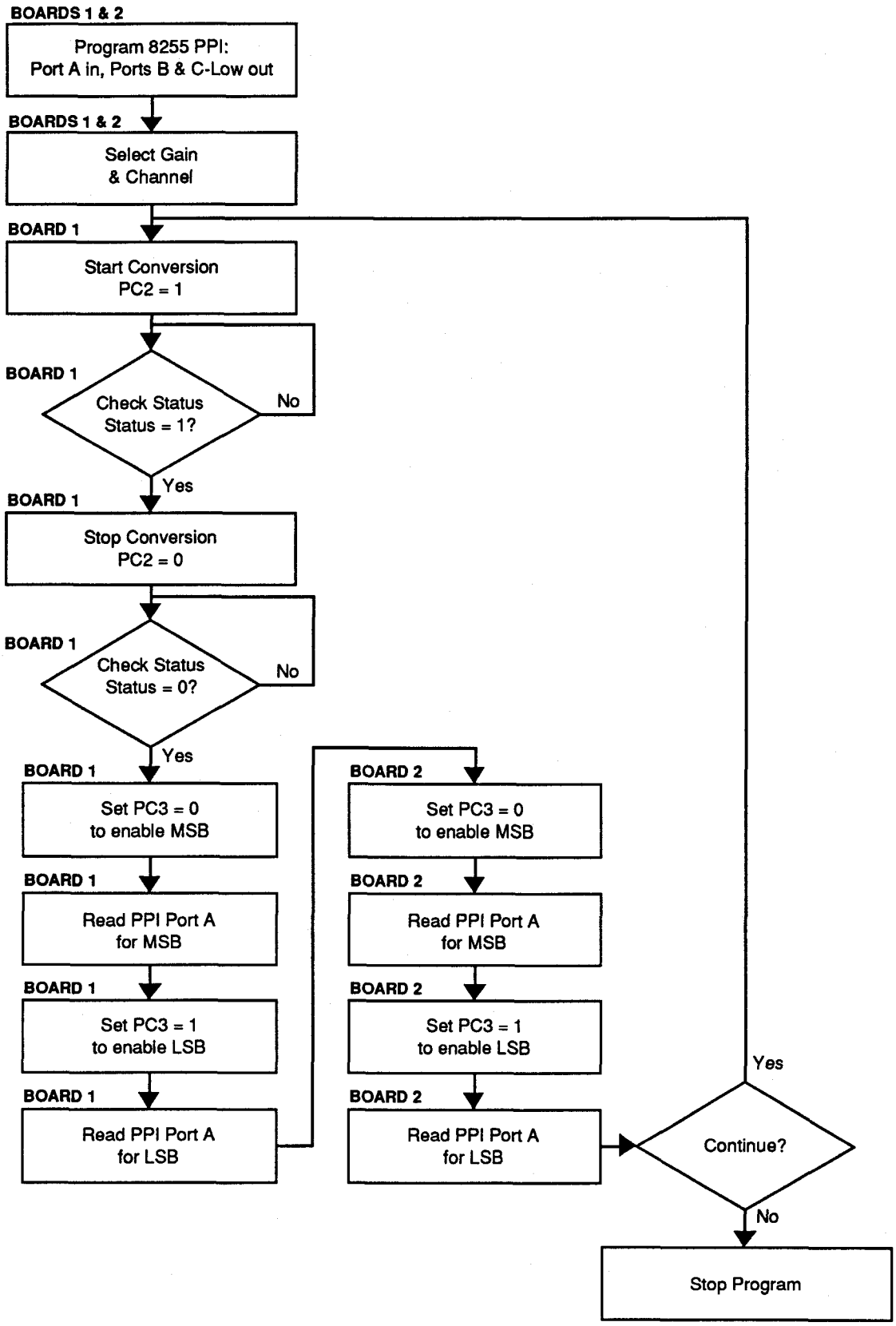


Fig. 4-7 — Cascaded Boards, Single Convert Mode Flow Diagram

Defining the I/O Map

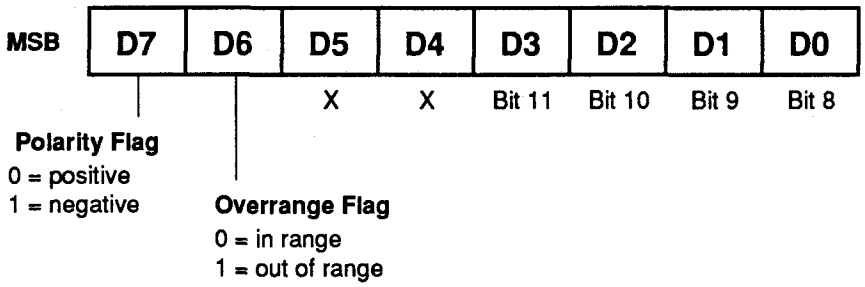
The I/O map for the AD510 is shown in Table 4-1. The base address (designated as BA) can be selected by setting the jumper on P2, as described in Chapter 1, *Board Settings*. One of eight starting address locations can be selected. The board is shipped with the base address set at 300 hex (768 decimal). The following sections describe the register contents of each address used in the I/O map.

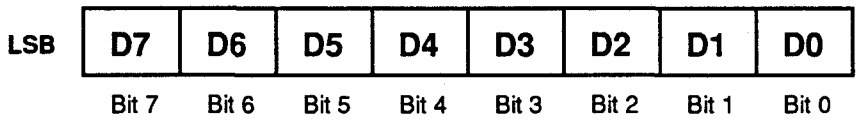
Table 4-1 — AD510 I/O Map

Register Description	Read Function	Write Function	Address * (Decimal)
PPI Port A (Read Data)	Read A/D converted data, MSB & LSB	Not used	BA + 0
PPI Port B/Channel Select	Read selected channel (bits 0-2 only)	Program 5 digital output lines and selected channel	BA + 1
PPI Port C (PC0-3 are reserved for control)	Read PC4-PC7 digital input lines	Program PC0-PC3 control lines & PC4-PC7 digital output lines	BA + 2
PPI Control Word	Read PPI control settings	Program PPI configuration	BA + 3
Status	Read A/D converter status	Not used	BA + 4
* BA = Base Address			

BA + 0: PPI Port A — Read A/D Data (Read Only)

This address is used to read the MSB and LSB of the A/D conversion, as defined below. When PC3 of the PPI port C word is 0, the MSB is read; when PC3 is 1, the LSB is read. Bit 7 of the MSB tells you whether the converted voltage is a positive or negative value. Bit 6 goes to 1 when the input voltage exceeds the input voltage range. The table below shows the expected converted data for maximum positive and negative input voltages and for 0 volts.



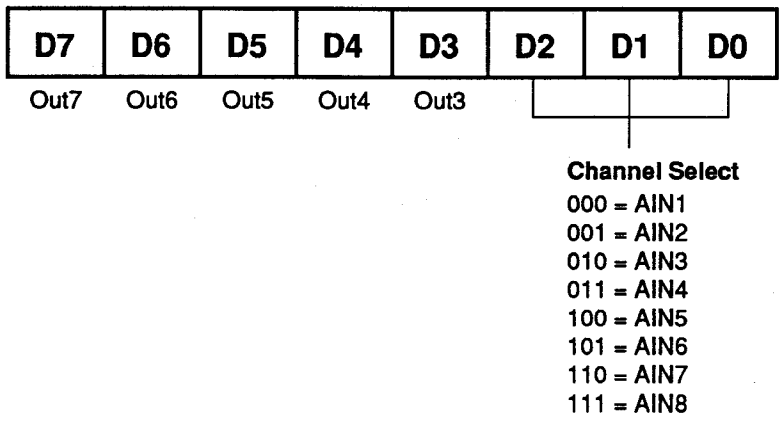


Polarity Flag	Converted Data (12 bits)	Voltage
0	1111 1111 1111	+4.99878 volts
1	1111 1111 1111	-4.99878 volts
0	0000 0000 0000	0.0 volts

BA + 1: PPI Port B — Channel Select/Digital Out (Read/Write)

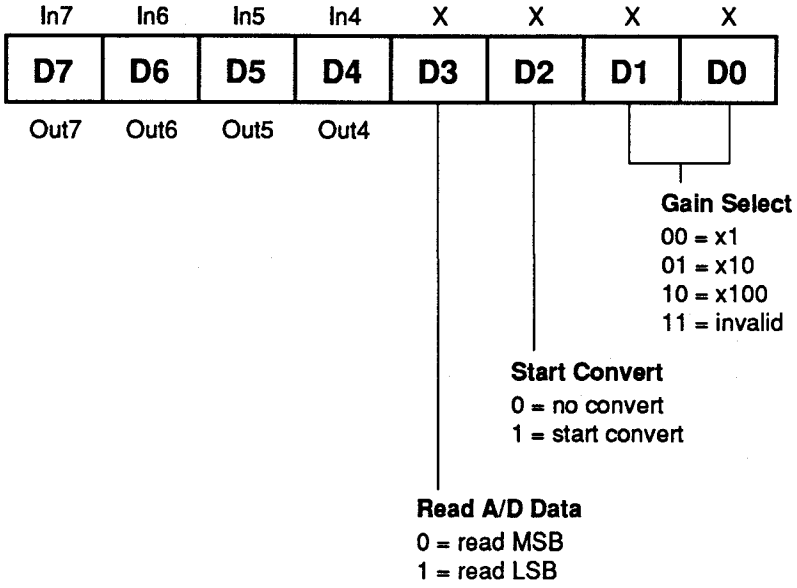
BA + 1 programs all eight PPI port B lines and reads the data on PB0-2. PB0, PB1, and PB2, the lower three bits, are used to program the multiplexer channel selection. Five bits of digital output data can be transferred between the board and an external device using PB3-7. Note that this port must be configured as output only, since PB0-2 must be output lines.

A write selects the channel on PB0-2 and transfers the written data from PB3-7 through P4 to an external device. Whenever you change the channel selection, you must be careful to preserve the digital output line settings, and vice versa, as necessary for your application. If you are writing to the channel bits and want to preserve the data in PB3-7, you can do this by first reading the port B data and then ANDing its contents with hex F8 (decimal 248). This resets PB0-2 while preserving PB3-8. Then, OR this result with the desired channel (PB3-8 must be 0 in your channel select word) and write the data back to port B. If you are writing to the digital output lines and want to preserve your channel setting, you can do this by first reading the port B data and ANDing its contents with hex 07 (decimal 7). This resets PB3-7 while preserving PB0-2. Then, OR this result with the data you want to output on PB3-7 (PB0-2 must be 0 in your digital output word) and write the data back to port B.



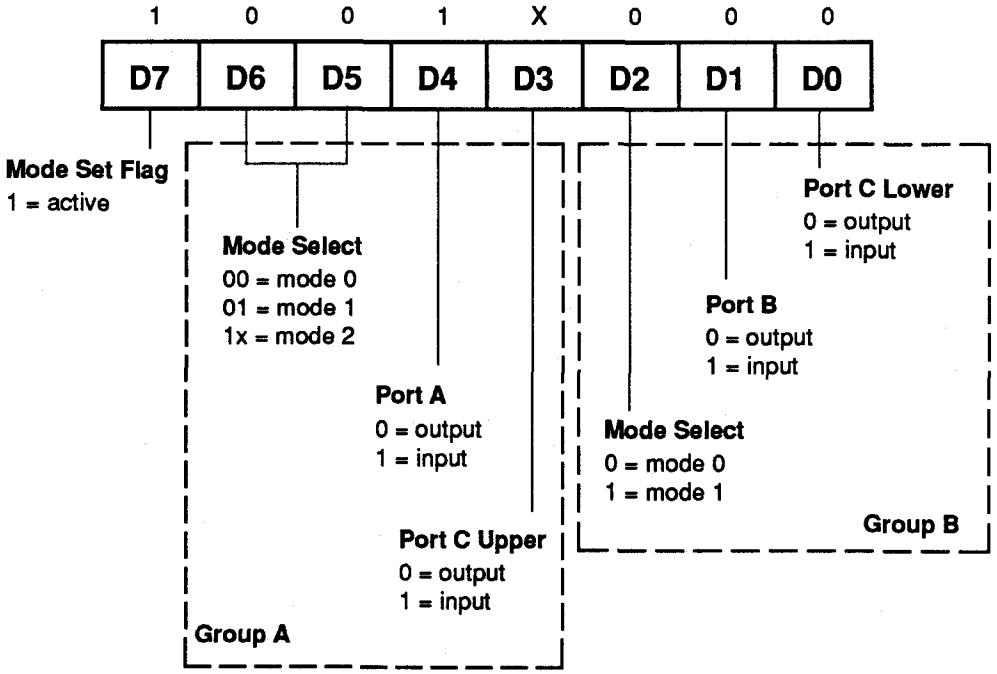
BA + 2: PPI Port C — A/D Control/Digital I/O (Read/Write)

BA + 2 programs all eight PPI port C lines and reads the data on the port C upper lines. PC0 through PC3 are used to program the gain and to control the A/D converter. PC4 through PC7 transfer digital input and digital output data between the board and an external device. A read transfers data from the external device through P4 and into PC4-7; a write transfers the written data from PC4-7 through P4 to an external device.

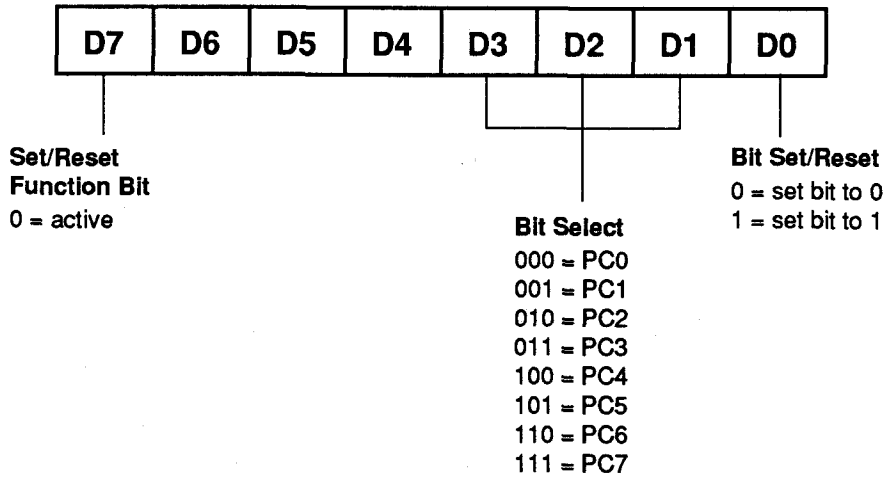


BA + 3: PPI Control Word (Read/Write)

When bit 7 of this word is set to 1, a write programs the PPI configuration and a read shows the PPI settings:

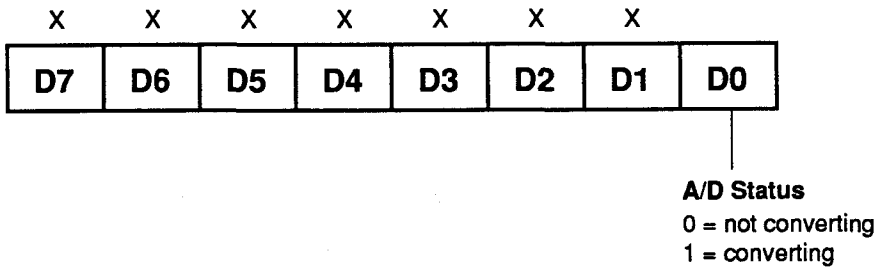


When bit 7 of this word is set to 0, a write can be used to individually program the port C lines:



BA + 4: Status (Read Only)

A read provides the A/D converter status bit defined below. When you start a conversion, the status line goes high. After the line goes high, then watch for it to go low, signaling that the conversion is complete.



Example Programs

The software included with your AD510 board contains example programs in BASIC, Turbo Pascal, and Turbo C to help you get started using the board. Source code is also included.

The AD510 board is easily programmed from any high-level language's I/O reference instructions. In BASIC, these instructions are INP and OUT; in Turbo Pascal, they are port and port; and in Turbo C, they are inportb and outportb.

Before using the software included with your board, make a backup copy of the disk. You may make as many backups as you need. The example programs included on the disk are listed below.

BASIC Programs:

SCAN.BAS	8-channel scanning
GAINS.BAS	changing the gain and performing A/D conversions
READ.BAS	performing A/D conversions
MVOLTS.BAS	performing A/D conversions in millivolts

Turbo Pascal Programs:

ADC.PSL	performing a single A/D conversion
MUX.PSL	selecting the analog input channel
PGA.PSL	setting the gain
INIT.PSL	initializing the board
MVOLTS.PAS	performing A/D conversions in millivolts
AUTO.PAS	autoranging the DVM
DEMO.PAS	a simple A/D conversion program
SCAN.PAS	8-channel scanning

Turbo C Programs:

MUX.H	selecting the analog input channel
INIT.H	initializing the board
ADC.H	performing a single A/D conversion
PGA.H	setting the gain
READ.C	performing A/D conversions
MVOLTS.C	performing A/D conversions in millivolts
SCAN.C	8-channel scanning
SIMPLE.C	a simple A/D conversion program

Lexicon Test and Diagnostics Tool

An interactive data collection and plotting utility program, Lexicon, is provided as a test and diagnostics tool. This program is described in Appendix D.

CHAPTER 5

CALIBRATION

This chapter tells you how to calibrate the AD510 using the four trimpots on the board. These trimpots calibrate the A/D input voltage range and gain. Calibration may be required if you change the A/D conversion rate from 7.5 to 30 Hz, or whenever you suspect inaccurate readings.

This chapter tells you how to calibrate the A/D converter gain and input voltage range. The offset and full-scale performance of the board's A/D converter is factory-calibrated for operation at 7.5 Hz. If you change the conversion rate, you may need to recalibrate your board. Any time you suspect inaccurate readings, you can check the accuracy of your conversions using the procedure below, and make adjusts as necessary.

Calibration is done with the board installed in your PC. You can access the trimpots with the computer's cover removed. Power up the computer and let the board circuitry stabilize for 15 minutes before you start calibrating.

Required Equipment

The following equipment is required for calibration:

- Precision Voltage Source: 0 to ±5 volts
- Digital Voltmeter: 5-1/2 digits
- Small Screwdriver (for trimpot adjustment)

A voltage source can be made using a 9-volt battery and a precision 10-turn trimpot as shown in Figure 5-1.

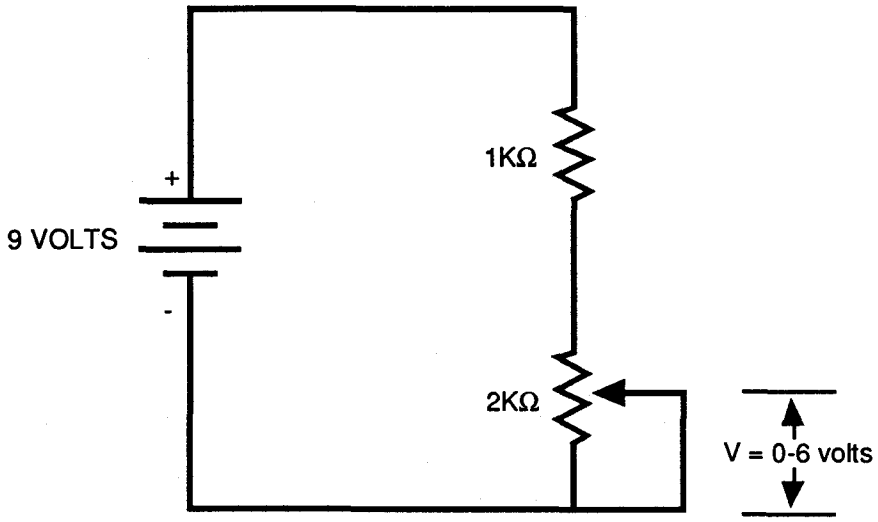


Fig. 5-1 — Adjustable Voltage Source

Figure 5-2 shows the board layout. The four trimpots used for calibration are shaded in the upper right area of the board.

A/D Calibration

During this procedure, you will make connections to the analog input at external I/O connector P4. The pin assignments for P4 are given in Appendix B.

Two adjustments are made to calibrate the A/D converter. One is the offset adjustment, and the other is the full-scale adjustment. Trimpots TR2 through TR4 adjust the offset, and TR1 adjust the full-scale voltage.

Offset Adjustment

The offset adjustment compensates for the inherent offset output voltage of the programmable gain amplifier. Three adjustments must made, one for each gain setting.

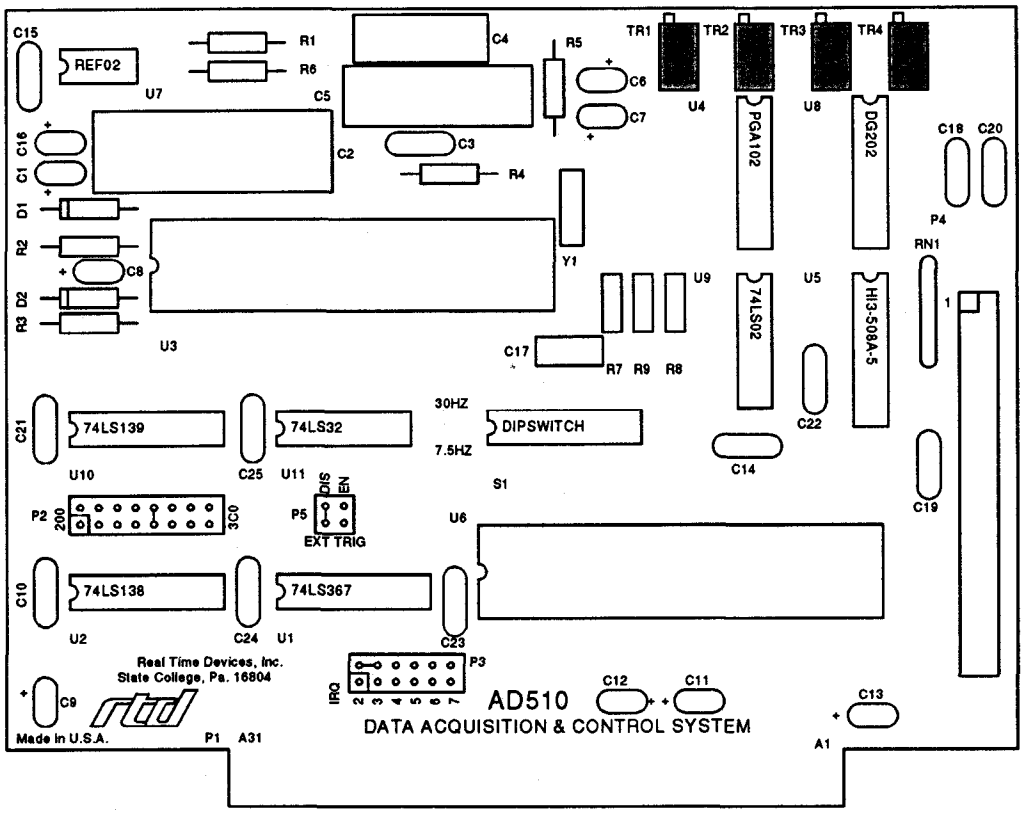


Fig. 5-2 — Board Layout Showing Calibration Trimpots

To adjust the offset error, connect 0 volts to analog input channel AIN1 (this can be done by simply connecting the analog input to ground). Set the gain to 1, and start continuous A/D conversions. Adjust trimpot TR2 until the conversion output is zero. Then change the gain to 10 and repeat the conversion process, adjusting TR3 until the conversion output is zero. Finally, change the gain to 100, and repeat the procedure, this time adjusting TR4 until the output is zero.

The trimpot assignment is summarized below:

Gain	PC1 Setting	PC0 Setting	Trimpot
x1	0	0	TR2
x10	1	0	TR3
x100	0	1	TR4

The offset voltage is related solely to the performance of the programmable gain amplifier; however, the offset can affect the rollover performance of the A/D converter if it is not set to zero. Rollover is the difference in conversion results between voltages having the same amplitude but different polarities. Any gains which do not have a zero offset will give readings that are shifted from an ideal zero reference. This means that the positive and negative readings for the same voltage will be slightly different. This difference is caused by the offset.

Full-Scale Adjustment

The full-scale adjustment calibrates the reference voltage used by the A/D converter to compensate for the analog input circuitry.

Set the gain to 1, the channel select to AIN1, and start continuous A/D conversions and display the conversion results. Apply the voltage source to AIN1 and adjust it so that it reads 4.9982 volts (4998.2 millivolts) (see Table 5-1). Adjust trimpot TR1 until the conversion data flickers between all 1s and 0 in the least significant bit place (hexadecimal FFF and FFE). Change the gain to 10, and repeat the procedure, this time using an input voltage of .49982 volts (499.82 millivolts). Finally, repeat the procedure, changing the gain to 100 and setting the input voltage at .044982 volts (49.982 millivolts).

Table 5-1 provides a reference for the ideal A/D converter input voltage for each bit weight in each voltage range. The first line is the ideal full-scale (all ones) value, and each successive line decreases by one bit weight. Between the full-scale voltage and the next lower bit weight is the voltage value for full-scale minus 1-1/2 bits. Note that the voltage values in the table are in millivolts.

Table 5-1 — A/D Converter Calibration Table

A/D Bit Weight	Ideal Input Voltage (in millivolts)		
	±5 V (gain = 1)	±0.5V (gain = 10)	±.05V (gain = 100)
4095 (Full Scale)	4998.8	499.88	49.987
FS = 1-1/2 LSB	4998.2	499.82	49.982
2048	2500.0	250.00	25.000
1024	1250.0	125.00	12.500
512	625.00	62.500	6.2500
256	312.50	31.250	3.1250
128	156.250	15.6250	1.5625
64	78.125	7.8125	0.7813
32	39.063	3.9063	0.3906
16	19.5313	1.9531	0.1953
8	9.7656	0.9766	0.0977
4	4.8828	0.4883	0.0488
2	2.4414	0.2441	0.0244
1	1.2207	0.1221	0.0122

APPENDIX A

AD510 SPECIFICATIONS

AD510 Characteristics Typical @ 25° C

Interface

IBM PC/XT/AT compatible
Jumper-selectable base address, I/O mapped
Jumper-selectable interrupts

Analog Input

8 single-ended inputs
Input impedance ..... >10 megohms
Gains, software selectable 1,10, 100
Gain error ..... 0.5%, typ; 1%, max
Input range ..... ±5, ±0.5, or ±0.05 volts
Overvoltage protection ..... ±35 Vdc
Settling time ..... 5 µsec, max

A/D Converter ICL7109

Type ..... Dual slope integrating with auto-zero
Resolution ..... 12 bits plus sign
Conversion rate ..... 7.5/30 Hz, switch-selectable
Relative accuracy (gain = 1) ..... ±1 bit
Linearity (gain = 1) ..... ±1 bit (7.5 Hz), ±3 bits (30 Hz)
Rollover error ..... ±1 bit

Digital I/O ..... CMOS 82C55 (Optional NMOS 8255)

Number of lines ..... 9 (4 input or output, 5 output)

Miscellaneous Outputs (PC bus-sourced)

Reset output
±12 volts
Digital ground

Current Requirements

+5 volts ..... 14 mA
+12 volts ..... 9 mA
–12 volts ..... 7 mA

Connector

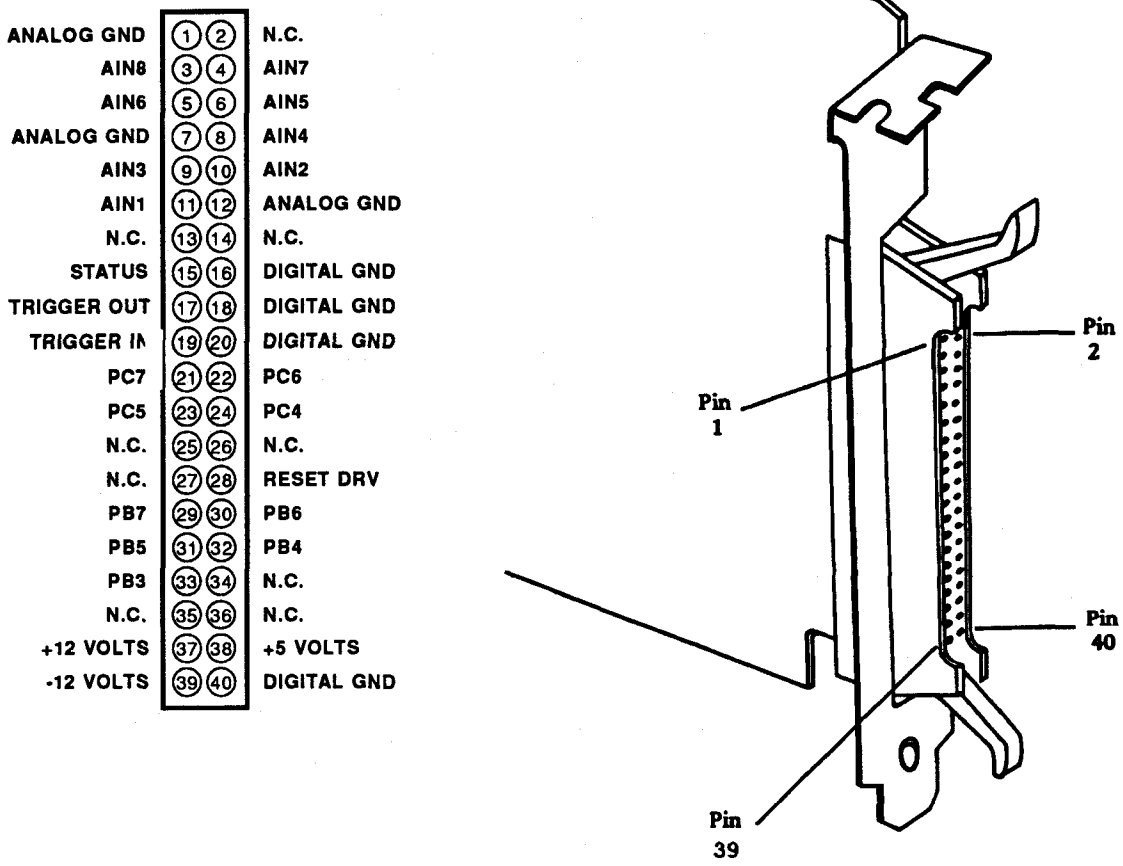
40-pin, right angle, shrouded header with ejector tabs

Size

Short slot — 3.875"H x 5.25"W (99mm x 134mm)

APPENDIX B

CONNECTOR PIN ASSIGNMENTS


ANALOG GND 1 2 N.C.
AIN8 3 4 AIN7
AIN6 5 6 AIN5
ANALOG GND 7 8 AIN4
AIN3 9 10 AIN2
AIN1 11 12 ANALOG GND
N.C. 13 14 N.C.
STATUS 15 16 DIGITAL GND
TRIGGER OUT 17 18 DIGITAL GND
TRIGGER IN 19 20 DIGITAL GND
PC7 21 22 PC6
PC5 23 24 PC4
N.C. 25 26 N.C.
N.C. 27 28 RESET DRV
PB7 29 30 PB6
PB5 31 32 PB4
PB3 33 34 N.C.
N.C. 35 36 N.C.
+12 VOLTS 37 38 +5 VOLTS
-12 VOLTS 39 40 DIGITAL GND
Pin 1
Pin 2
Pin 40
Pin 39

AD510 P4 Connector/Mating Connector		
Manufacturer	**AD510 P4 Connector**	**P4 Mating Connector**
Fujitsu 3M Robinson Nugent MIL C-83503	FCN-705Q040-AU/M	FCN-707B040-AU/B 3417-7040 IDS-C40PK-C-SR-TG M83503/7-09

APPENDIX C

COMPONENT DATA SHEETS

Intel 82C55A Programmable Peripheral Interface
Data Sheet Reprint

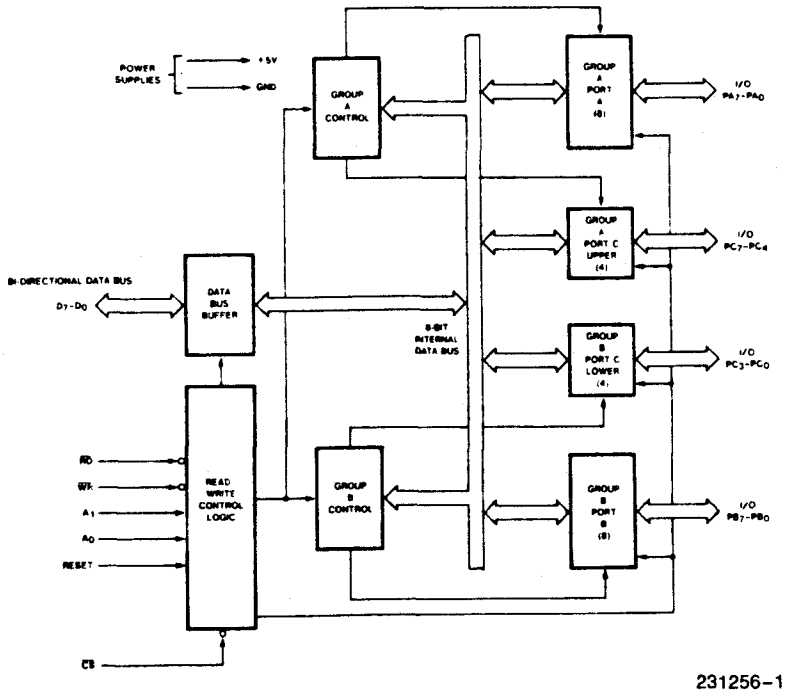
82C55A
CHMOS PROGRAMMABLE PERIPHERAL INTERFACE

- **Compatible with all Intel and Most Other Microprocessors**
- **High Speed, "Zero Wait State" Operation with 8 MHz 8086/88 and 80186/188**
- **24 Programmable I/O Pins**
- **Low Power CHMOS**
- **Completely TTL Compatible**
- **Control Word Read-Back Capability**
- **Direct Bit Set/Reset Capability**
- **2.5 mA DC Drive Capability on all I/O Port Outputs**
- **Available in 40-Pin DIP and 44-Pin PLCC**
- **Available in EXPRESS**
  - **— Standard Temperature Range**
  - **— Extended Temperature Range**

The Intel 82C55A is a high-performance, CHMOS version of the industry standard 8255A general purpose programmable I/O device which is designed for use with all Intel and most other microprocessors. It provides 24 I/O pins which may be individually programmed in 2 groups of 12 and used in 3 major modes of operation. The 82C55A is pin compatible with the NMOS 8255A and 8255A-5.

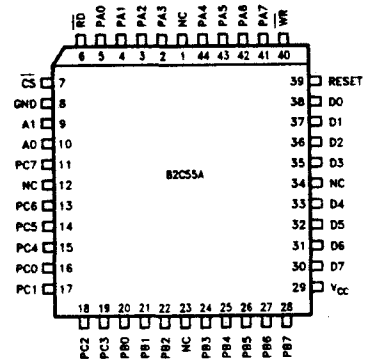
In MODE 0, each group of 12 I/O pins may be programmed in sets of 4 and 8 to be inputs or outputs. In MODE 1, each group may be programmed to have 8 lines of input or output. 3 of the remaining 4 pins are used for handshaking and interrupt control signals. MODE 2 is a strobed bi-directional bus configuration.

The 82C55A is fabricated on Intel's advanced CHMOS III technology which provides low power consumption with performance equal to or greater than the equivalent NMOS product. The 82C55A is available in 40-pin DIP and 44-pin plastic leaded chip carrier (PLCC) packages.

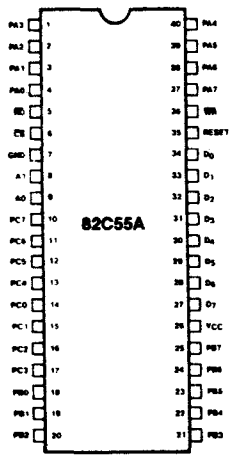


231256-1

Figure 1. 82C55A Block Diagram



231256-31



231256-2

Figure 2. 82C55A Pinout

Diagrams are for pin reference only. Package sizes are not to scale.

September 1987
Order Number: 231256-004

Table 1. Pin Description

Symbol	Pin Number Dip	Pin Number PLCC	Type	Name and Function
PA_{3-0}	1-4	2-5	I/O	**PORT A, PINS 0-3:** Lower nibble of an 8-bit data output latch/ buffer and an 8-bit data input latch.
$\overline{RD}$	5	6	I	**READ CONTROL:** This input is low during CPU read operations.
$\overline{CS}$	6	7	I	**CHIP SELECT:** A low on this input enables the 82C55A to respond to $\overline{RD}$ and $\overline{WR}$ signals. $\overline{RD}$ and WR are ignored otherwise.
GND	7	8		**System Ground**
A_{1-0}	8-9	9-10	I	**ADDRESS:** These input signals, in conjunction $\overline{RD}$ and $\overline{WR}$, control the selection of one of the three ports or the control word registers.

A_1	A_0	$\overline{RD}$	$\overline{WR}$	$\overline{CS}$	Input Operation (Read)
0	0	0	1	0	Port A - Data Bus
0	1	0	1	0	Port B - Data Bus
1	0	0	1	0	Port C - Data Bus
1	1	0	1	0	Control Word - Data Bus
					Output Operation (Write)
0	0	1	0	0	Data Bus - Port A
0	1	1	0	0	Data Bus - Port B
1	0	1	0	0	Data Bus - Port C
1	1	1	0	0	Data Bus - Control
					Disable Function
X	X	X	X	1	Data Bus - 3 - State
X	X	1	1	0	Data Bus - 3 - State

PC_{7-4}	10-13	11,13-15	I/O	**PORT C, PINS 4-7:** Upper nibble of an 8-bit data output latch/ buffer and an 8-bit data input buffer (no latch for input). This port can be divided into two 4-bit ports under the mode control. Each 4-bit port contains a 4-bit latch and it can be used for the control signal outputs and status signal inputs in conjunction with ports A and B.
PC_{0-3}	14-17	16-19	I/O	**PORT C, PINS 0-3:** Lower nibble of Port C.
PB_{0-7}	18-25	20-22, 24-28	I/O	**PORT B, PINS 0-7:** An 8-bit data output latch/buffer and an 8-bit data input buffer.
V_{CC}	26	29		**SYSTEM POWER:** + 5V Power Supply.
D_{7-0}	27-34	30-33, 35-38	I/O	**DATA BUS:** Bi-directional, tri-state data bus lines, connected to system data bus.
RESET	35	39	I	**RESET:** A high on this input clears the control register and all ports are set to the input mode.
$\overline{WR}$	36	40	I	**WRITE CONTROL:** This input is low during CPU write operations.
PA_{7-4}	37-40	41-44	I/O	**PORT A, PINS 4-7:** Upper nibble of an 8-bit data output latch/ buffer and an 8-bit data input latch.
NC		1, 12, 23, 34		No Connect

82C55A FUNCTIONAL DESCRIPTION

General

The 82C55A is a programmable peripheral interface device designed for use in Intel microcomputer systems. Its function is that of a general purpose I/O component to interface peripheral equipment to the microcomputer system bus. The functional configuration of the 82C55A is programmed by the system software so that normally no external logic is necessary to interface peripheral devices or structures.

Data Bus Buffer

This 3-state bidirectional 8-bit buffer is used to interface the 82C55A to the system data bus. Data is transmitted or received by the buffer upon execution of input or output instructions by the CPU. Control words and status information are also transferred through the data bus buffer.

Read/Write and Control Logic

The function of this block is to manage all of the internal and external transfers of both Data and Control or Status words. It accepts inputs from the CPU Address and Control busses and in turn, issues commands to both of the Control Groups.

Group A and Group B Controls

The functional configuration of each port is programmed by the systems software. In essence, the CPU "outputs" a control word to the 82C55A. The control word contains information such as "mode", "bit set", "bit reset", etc., that initializes the functional configuration of the 82C55A.

Each of the Control blocks (Group A and Group B) accepts "commands" from the Read/Write Control Logic, receives "control words" from the internal data bus and issues the proper commands to its associated ports.

Control Group A - Port A and Port C upper (C7–C4)
Control Group B - Port B and Port C lower (C3–C0)

The control word register can be both written and read as shown in the address decode table in the pin descriptions. Figure 6 shows the control word format for both Read and Write operations. When the control word is read, bit D7 will always be a logic "1", as this implies control word mode information.

Ports A, B, and C

The 82C55A contains three 8-bit ports (A, B, and C). All can be configured in a wide variety of functional characteristics by the system software but each has its own special features or "personality" to further enhance the power and flexibility of the 82C55A.

Port A. One 8-bit data output latch/buffer and one 8-bit input latch buffer. Both "pull-up" and "pull-down" bus hold devices are present on Port A.

Port B. One 8-bit data input/output latch/buffer. Only "pull-up" bus hold devices are present on Port B.

Port C. One 8-bit data output latch/buffer and one 8-bit data input buffer (no latch for input). This port can be divided into two 4-bit ports under the mode control. Each 4-bit port contains a 4-bit latch and it can be used for the control signal outputs and status signal inputs in conjunction with ports A and B. Only "pull-up" bus hold devices are present on Port C.

See Figure 4 for the bus-hold circuit configuration for Port A, B, and C.

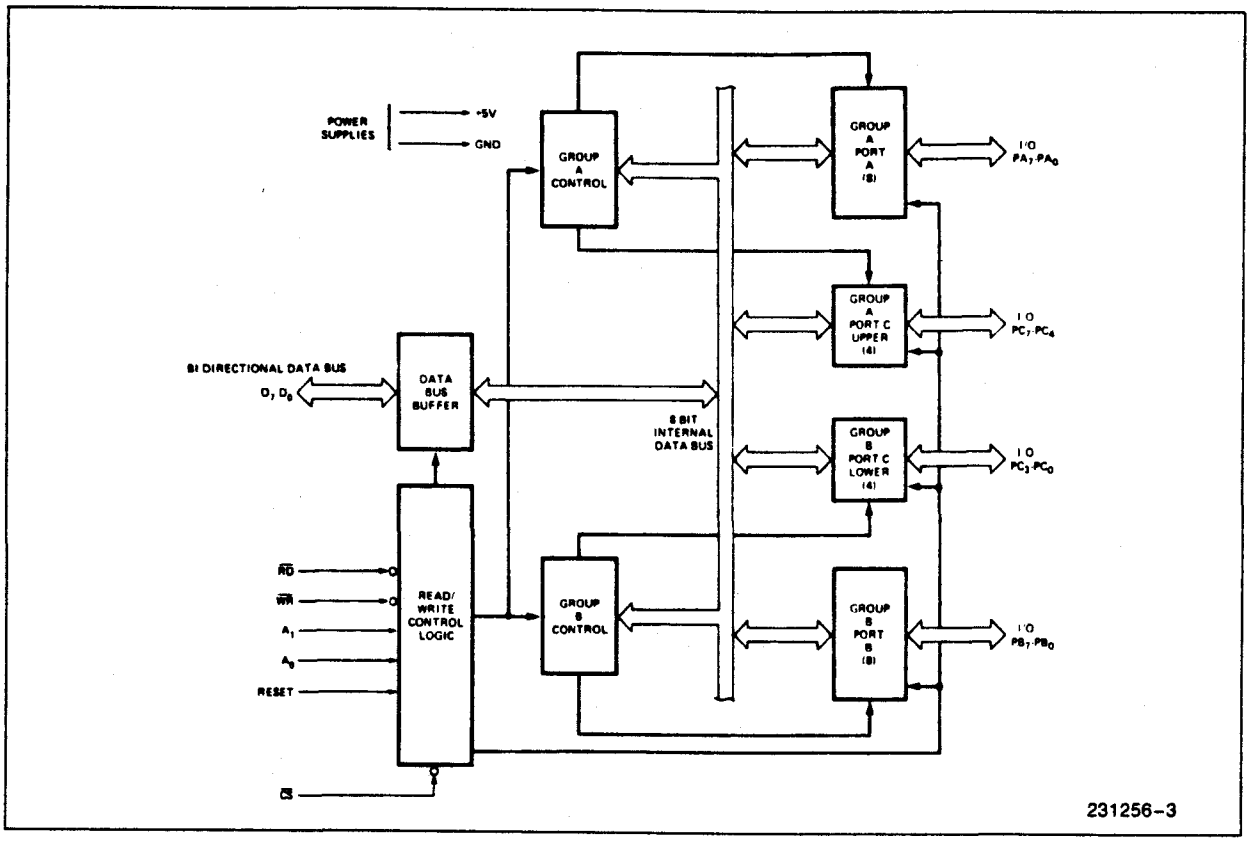
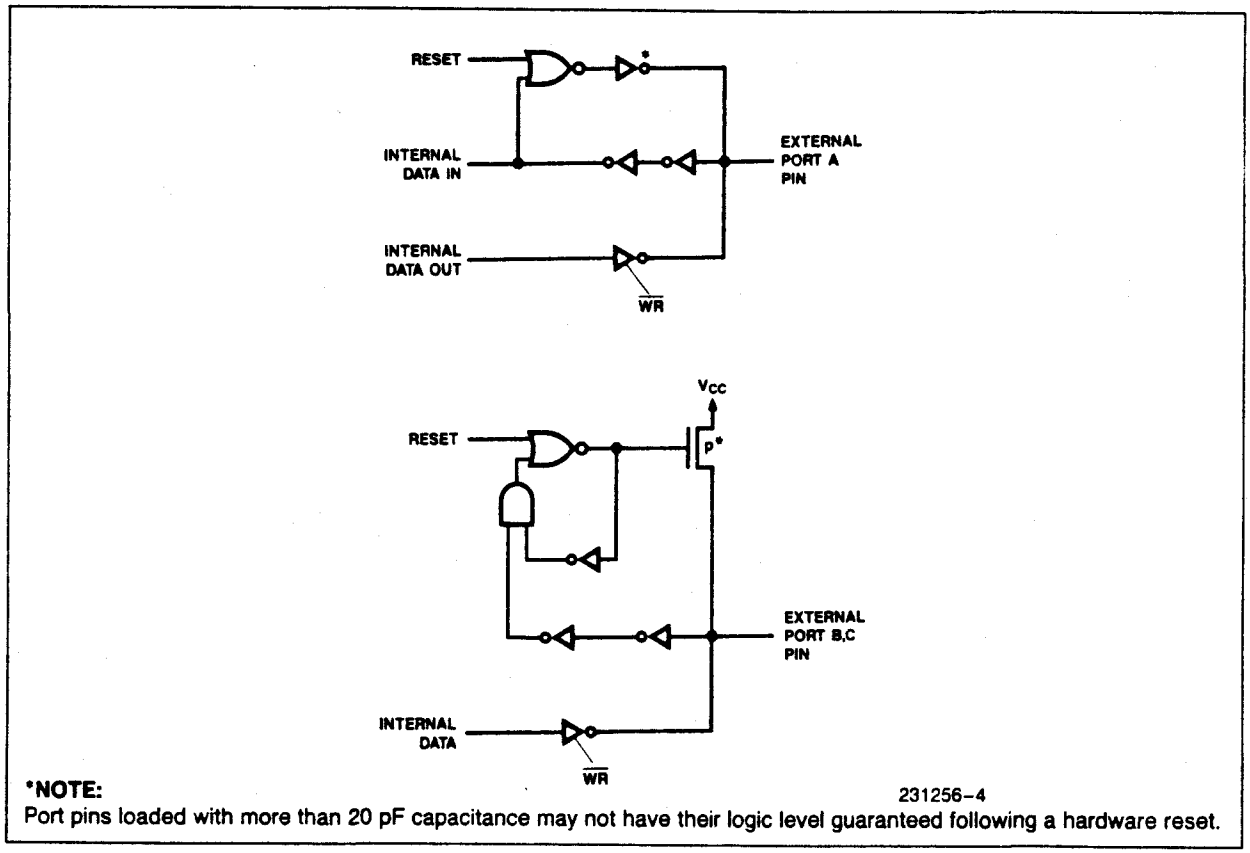


Figure 3. 82C55A Block Diagram Showing Data Bus Buffer and Read/Write Control Logic Functions



***NOTE:**
Port pins loaded with more than 20 pF capacitance may not have their logic level guaranteed following a hardware reset.

Figure 4. Port A, B, C, Bus-hold Configuration

82C55A OPERATIONAL DESCRIPTION

Mode Selection

There are three basic modes of operation that can be selected by the system software:

Mode 0 — Basic input/output
Mode 1 — Strobed Input/output
Mode 2 — Bi-directional Bus

When the reset input goes "high" all ports will be set to the input mode with all 24 port lines held at a logic "one" level by the internal bus hold devices (see Figure 4 Note). After the reset is removed the 82C55A can remain in the input mode with no additional initialization required. This eliminates the need for pullup or pulldown devices in "all CMOS" designs. During the execution of the system program, any of the other modes may be selected by using a single output instruction. This allows a single 82C55A to service a variety of peripheral devices with a simple software maintenance routine.

The modes for Port A and Port B can be separately defined, while Port C is divided into two portions as required by the Port A and Port B definitions. All of the output registers, including the status flip-flops, will be reset whenever the mode is changed. Modes may be combined so that their functional definition can be "tailored" to almost any I/O structure. For instance; Group B can be programmed in Mode 0 to monitor simple switch closings or display computational results. Group A could be programmed in Mode 1 to monitor a keyboard or tape reader on an interrupt-driven basis.

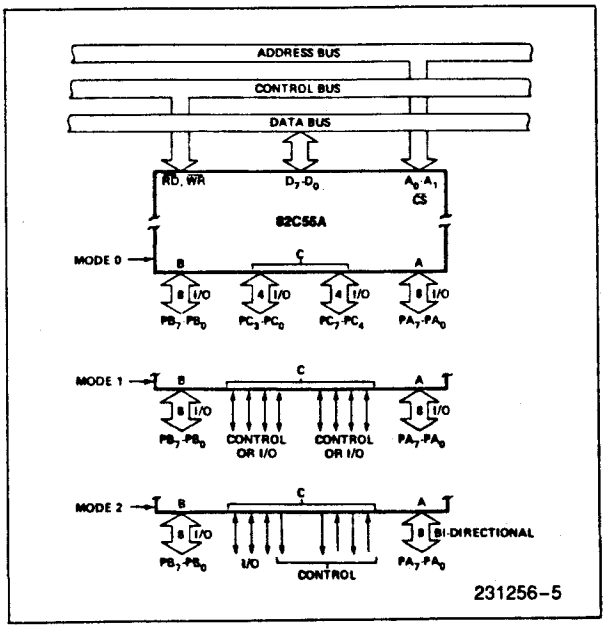


Figure 5. Basic Mode Definitions and Bus Interface

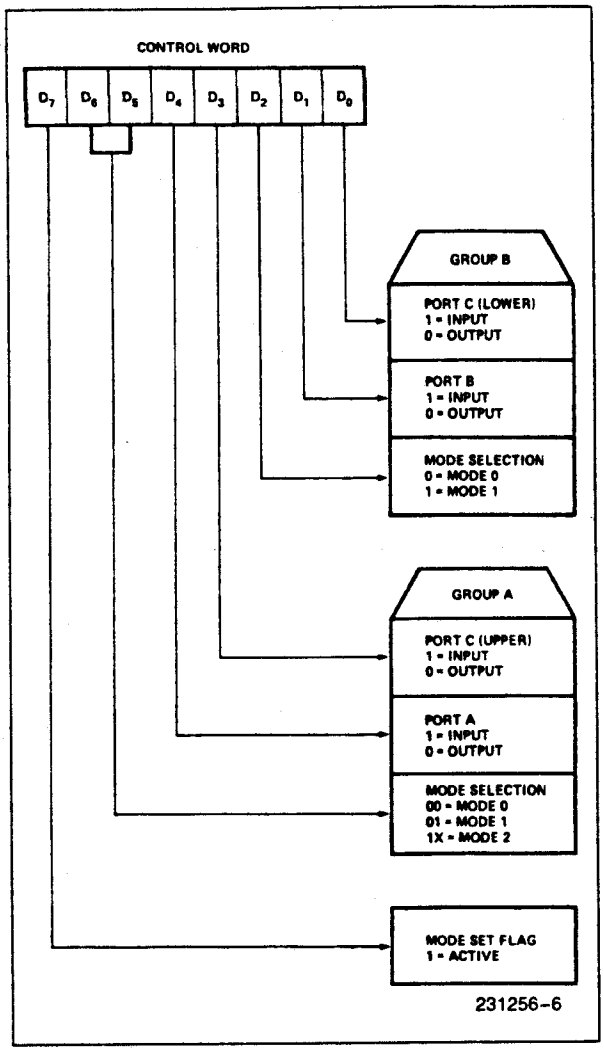


Figure 6. Mode Definition Format

The mode definitions and possible mode combinations may seem confusing at first but after a cursory review of the complete device operation a simple, logical I/O approach will surface. The design of the 82C55A has taken into account things such as efficient PC board layout, control signal definition vs PC layout and complete functional flexibility to support almost any peripheral device with no external logic. Such design represents the maximum use of the available pins.

Single Bit Set/Reset Feature

Any of the eight bits of Port C can be Set or Reset using a single OUTput instruction. This feature reduces software requirements in Control-based applications.

When Port C is being used as status/control for Port A or B, these bits can be set or reset by using the Bit Set/Reset operation just as if they were data output ports.

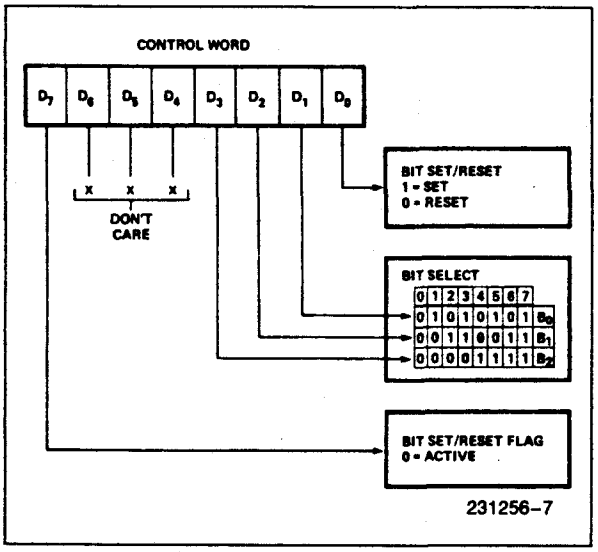

CONTROL WORD
D7 D6 D5 D4 D3 D2 D1 D0
X X X
DON'T CARE
BIT SET/RESET
1 = SET
0 = RESET
BIT SELECT
0 1 2 3 4 5 6 7
0 1 0 1 0 1 0 1 B0
0 0 1 1 0 0 1 1 B1
0 0 0 0 1 1 1 1 B2
BIT SET/RESET FLAG
0 = ACTIVE
231256-7

Figure 7. Bit Set/Reset Format

Interrupt Control Functions

When the 82C55A is programmed to operate in mode 1 or mode 2, control signals are provided that can be used as interrupt request inputs to the CPU. The interrupt request signals, generated from port C, can be inhibited or enabled by setting or resetting the associated INTE flip-flop, using the bit set/reset function of port C.

This function allows the Programmer to disallow or allow a specific I/O device to interrupt the CPU without affecting any other device in the interrupt structure.

INTE flip-flop definition:

(BIT-SET)—INTE is SET—Interrupt enable
(BIT-RESET)—INTE is RESET—Interrupt disable

Note:
All Mask flip-flops are automatically reset during mode selection and device Reset.

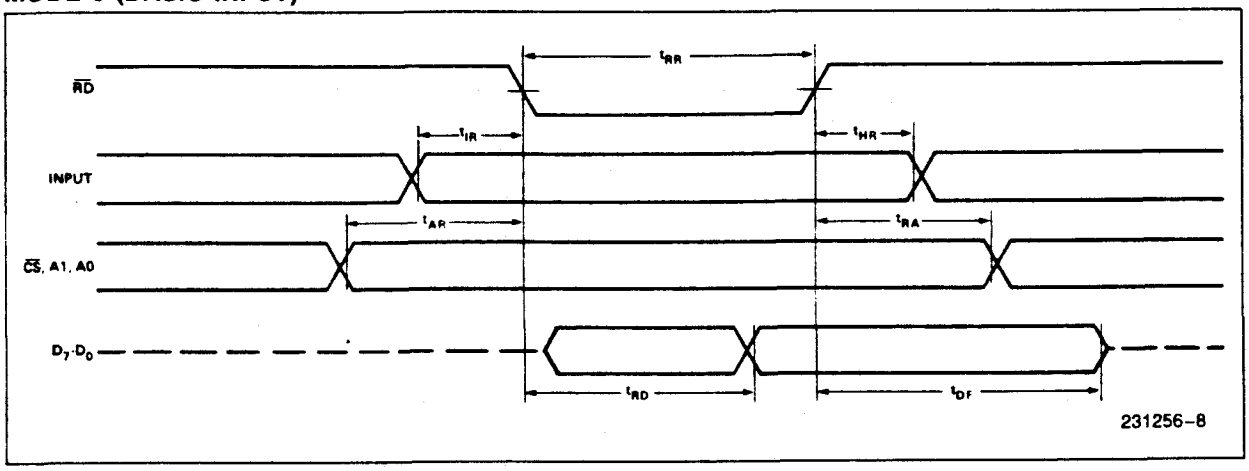
Operating Modes

Mode 0 (Basic Input/Output). This functional configuration provides simple input and output operations for each of the three ports. No "handshaking" is required, data is simply written to or read from a specified port.

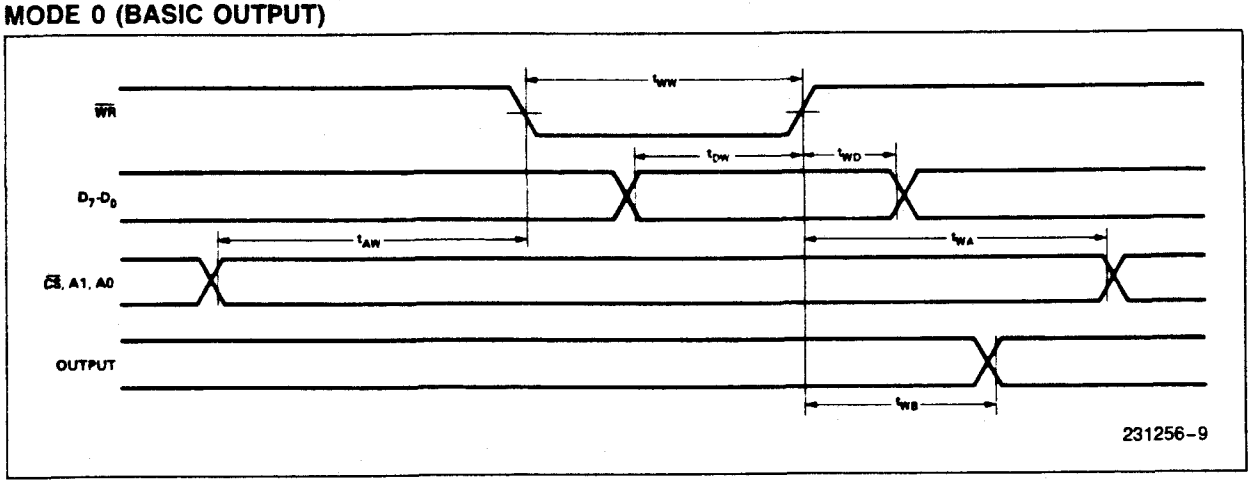
Mode 0 Basic Functional Definitions:

- Two 8-bit ports and two 4-bit ports.
- Any port can be input or output.
- Outputs are latched.
- Inputs are not latched.
- 16 different Input/Output configurations are possible in this Mode.

MODE 0 (BASIC INPUT)



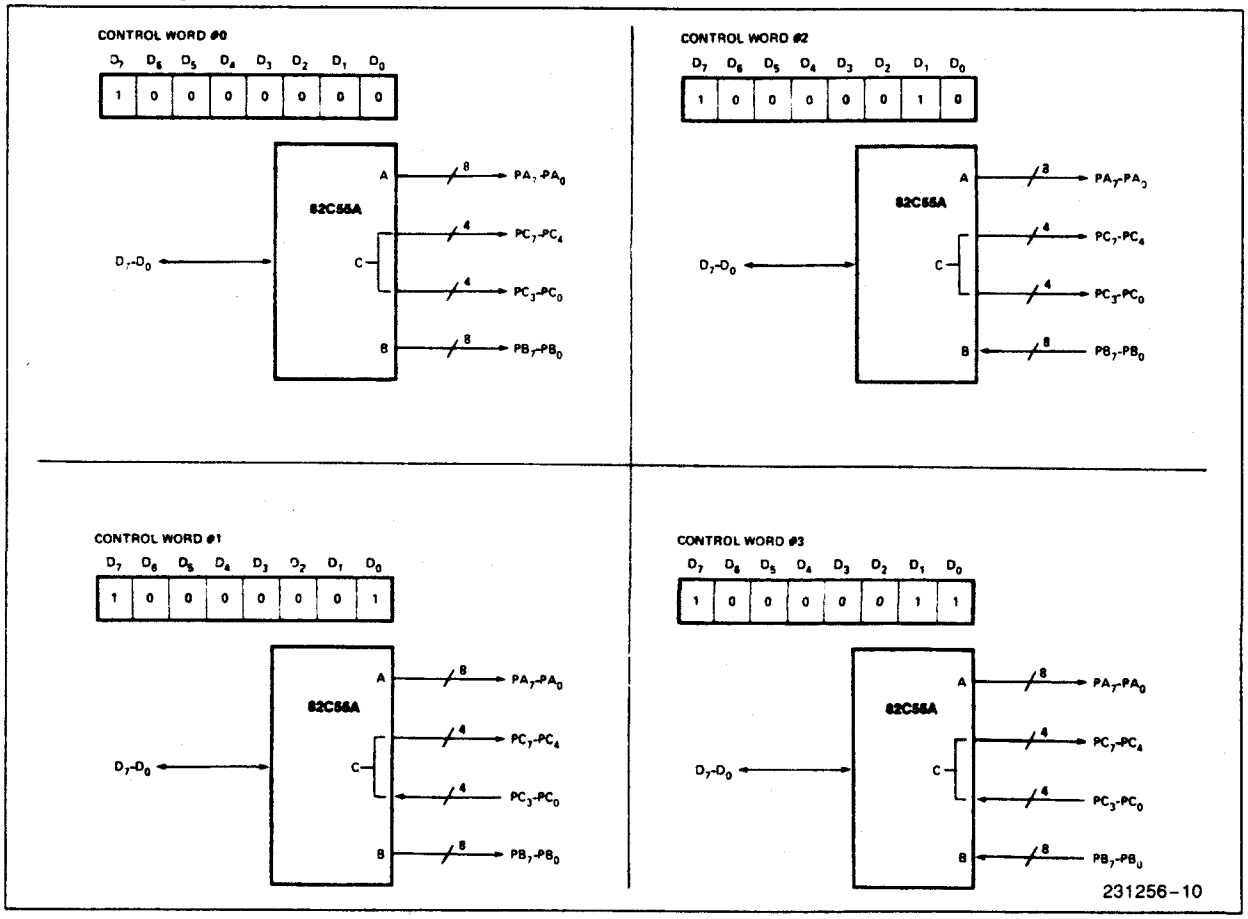
MODE 0 (BASIC OUTPUT)



MODE 0 Port Definition

A		B		GROUP A			GROUP B	
D_4	D_3	D_1	D_0	PORT A	PORT C (UPPER)	#	PORT B	PORT C (LOWER)
0	0	0	0	OUTPUT	OUTPUT	0	OUTPUT	OUTPUT
0	0	0	1	OUTPUT	OUTPUT	1	OUTPUT	INPUT
0	0	1	0	OUTPUT	OUTPUT	2	INPUT	OUTPUT
0	0	1	1	OUTPUT	OUTPUT	3	INPUT	INPUT
0	1	0	0	OUTPUT	INPUT	4	OUTPUT	OUTPUT
0	1	0	1	OUTPUT	INPUT	5	OUTPUT	INPUT
0	1	1	0	OUTPUT	INPUT	6	INPUT	OUTPUT
0	1	1	1	OUTPUT	INPUT	7	INPUT	INPUT
1	0	0	0	INPUT	OUTPUT	8	OUTPUT	OUTPUT
1	0	0	1	INPUT	OUTPUT	9	OUTPUT	INPUT
1	0	1	0	INPUT	OUTPUT	10	INPUT	OUTPUT
1	0	1	1	INPUT	OUTPUT	11	INPUT	INPUT
1	1	0	0	INPUT	INPUT	12	OUTPUT	OUTPUT
1	1	0	1	INPUT	INPUT	13	OUTPUT	INPUT
1	1	1	0	INPUT	INPUT	14	INPUT	OUTPUT
1	1	1	1	INPUT	INPUT	15	INPUT	INPUT

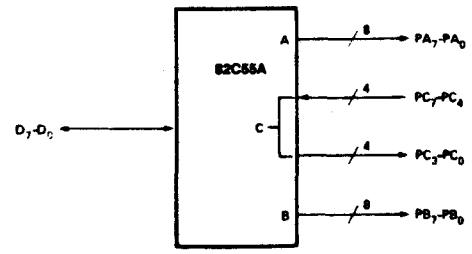
MODE 0 Configurations



MODE 0 Configurations (Continued)

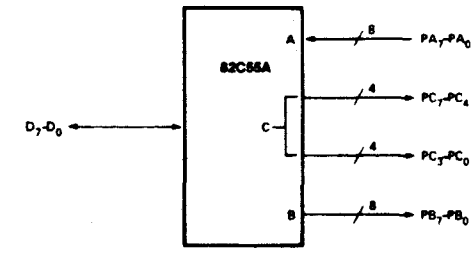
CONTROL WORD #4

D_7	D_6	D_5	D_4	D_3	D_2	D_1	D_0
1	0	0	0	1	0	0	0



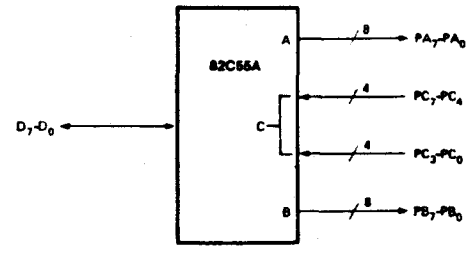
CONTROL WORD #8

D_7	D_6	D_5	D_4	D_3	D_2	D_1	D_0
1	0	0	1	0	0	0	0



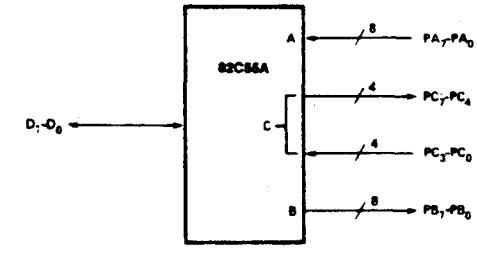
CONTROL WORD #5

D_7	D_6	D_5	D_4	D_3	D_2	D_1	D_0
1	0	0	0	1	0	0	1



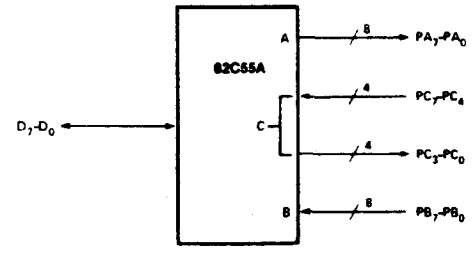
CONTROL WORD #9

D_7	D_6	D_5	D_4	D_3	D_2	D_1	D_0
1	0	0	1	0	0	0	1



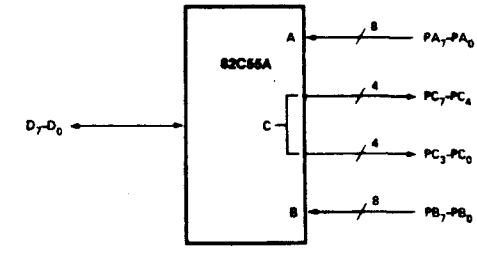
CONTROL WORD #6

D_7	D_6	D_5	D_4	D_3	D_2	D_1	D_0
1	0	0	0	1	0	1	0



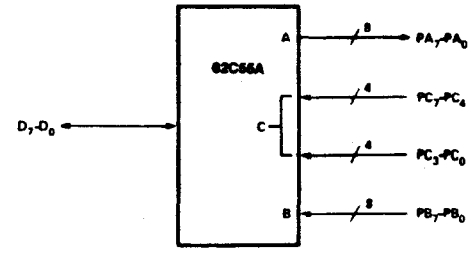
CONTROL WORD #10

D_7	D_6	D_5	D_4	D_3	D_2	D_1	D_0
1	0	0	1	0	0	1	0



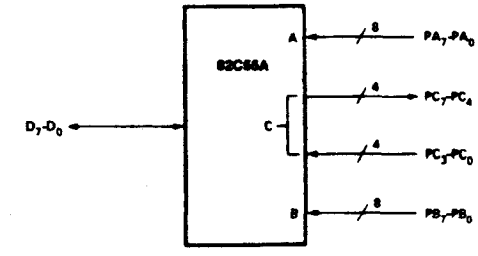
CONTROL WORD #7

D_7	D_6	D_5	D_4	D_3	D_2	D_1	D_0
1	0	0	0	1	0	1	1



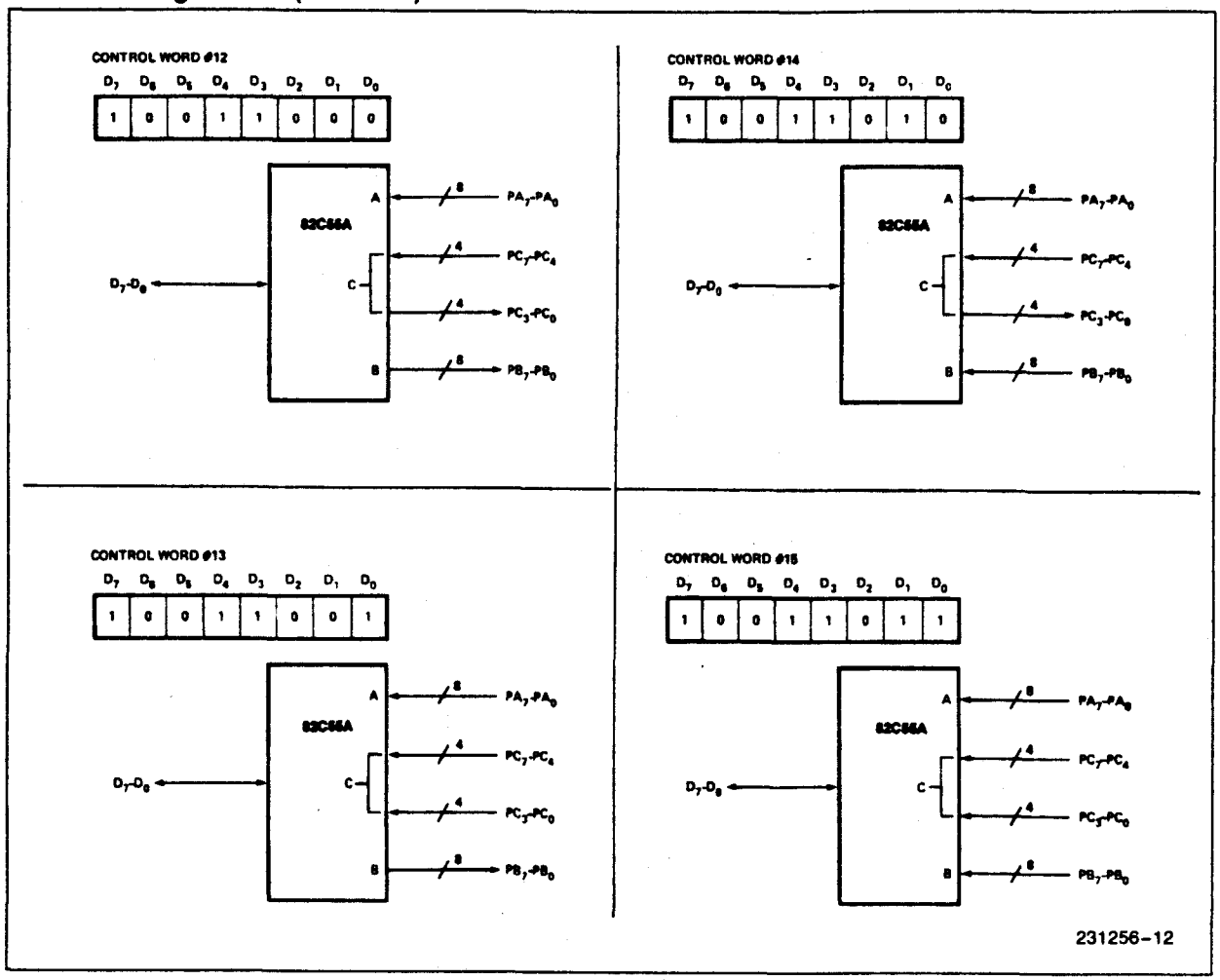
CONTROL WORD #11

D_7	D_6	D_5	D_4	D_3	D_2	D_1	D_0
1	0	0	1	0	0	1	1



231256-11

MODE 0 Configurations (Continued)



Operating Modes

MODE 1 (Strobed Input/Output). This functional configuration provides a means for transferring I/O data to or from a specified port in conjunction with strobes or "handshaking" signals. In mode 1, Port A and Port B use the lines on Port C to generate or accept these "handshaking" signals.

Mode 1 Basic functional Definitions:

- Two Groups (Group A and Group B).
- Each group contains one 8-bit data port and one 4-bit control/data port.
- The 8-bit data port can be either input or output Both inputs and outputs are latched.
- The 4-bit port is used for control and status of the 8-bit data port.

Input Control Signal Definition

$\overline{STB}$ (Strobe Input). A "low" on this input loads data into the input latch.

IBF (Input Buffer Full F/F)

A "high" on this output indicates that the data has been loaded into the input latch; in essence, an acknowledgement. IBF is set by $\overline{STB}$ input being low and is reset by the rising edge of the $\overline{RD}$ input.

INTR (Interrupt Request)

A "high" on this output can be used to interrupt the CPU when an input device is requesting service. INTR is set by the $\overline{STB}$ is a "one", IBF is a "one" and INTE is a "one". It is reset by the falling edge of $\overline{RD}$. This procedure allows an input device to request service from the CPU by simply strobing its data into the port.

INTE A

Controlled by bit set/reset of PC_4.

INTE B

Controlled by bit set/reset of PC_2.

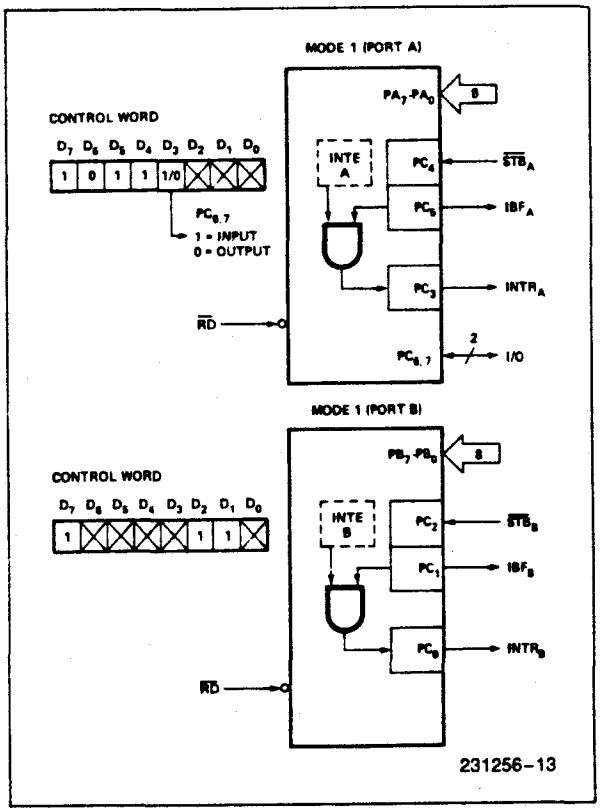


Figure 8. MODE 1 Input

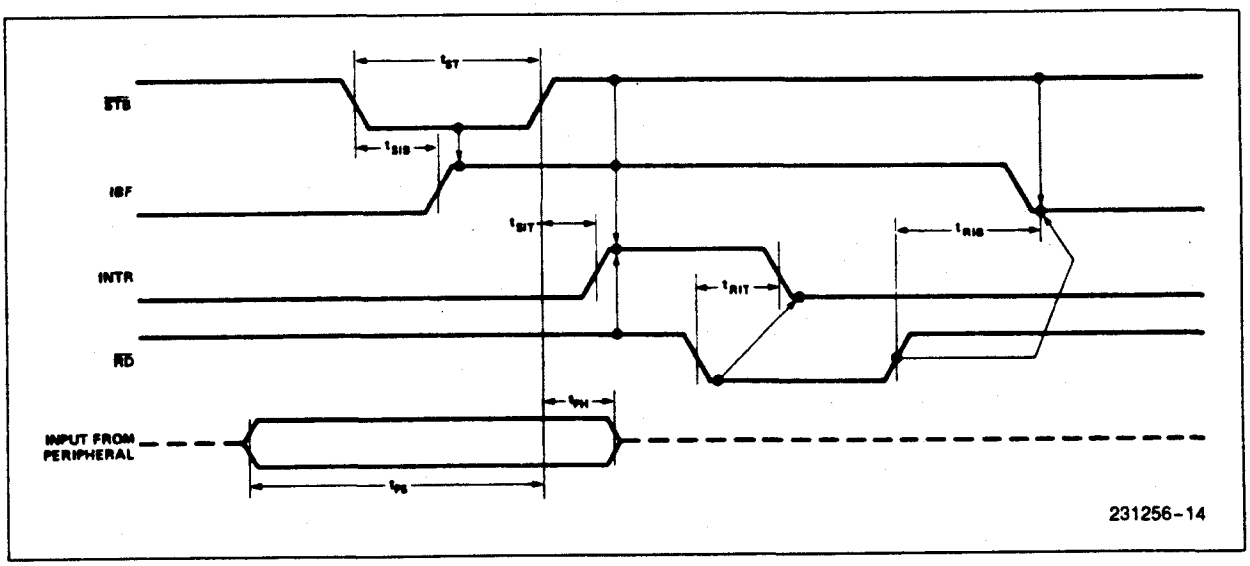


Figure 9. MODE 1 (Strobed Input)

Output Control Signal Definition

$\overline{OBF}$ (Output Buffer Full F/F). The $\overline{OBF}$ output will go "low" to indicate that the CPU has written data out to the specified port. The $\overline{OBF}$ F/F will be set by the rising edge of the $\overline{WR}$ input and reset by $\overline{ACK}$ Input being low.

$\overline{ACK}$ (Acknowledge Input). A "low" on this input informs the 82C55A that the data from Port A or Port B has been accepted. In essence, a response from the peripheral device indicating that it has received the data output by the CPU.

INTR (Interrupt Request). A "high" on this output can be used to interrupt the CPU when an output device has accepted data transmitted by the CPU. INTR is set when $\overline{ACK}$ is a "one", $\overline{OBF}$ is a "one" and INTE is a "one". It is reset by the falling edge of $\overline{WR}$.

INTE A

Controlled by bit set/reset of PC_6.

INTE B

Controlled by bit set/reset of PC_2.

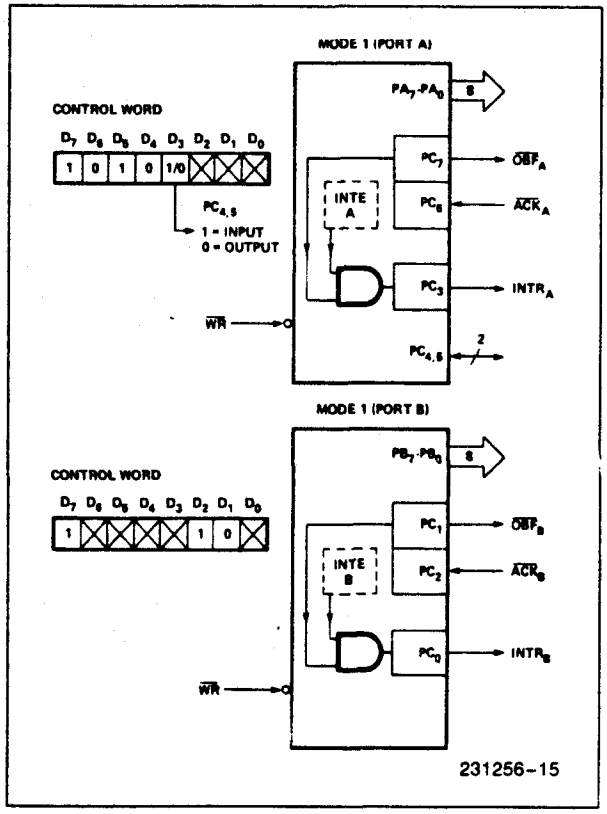


Figure 10. MODE 1 Output

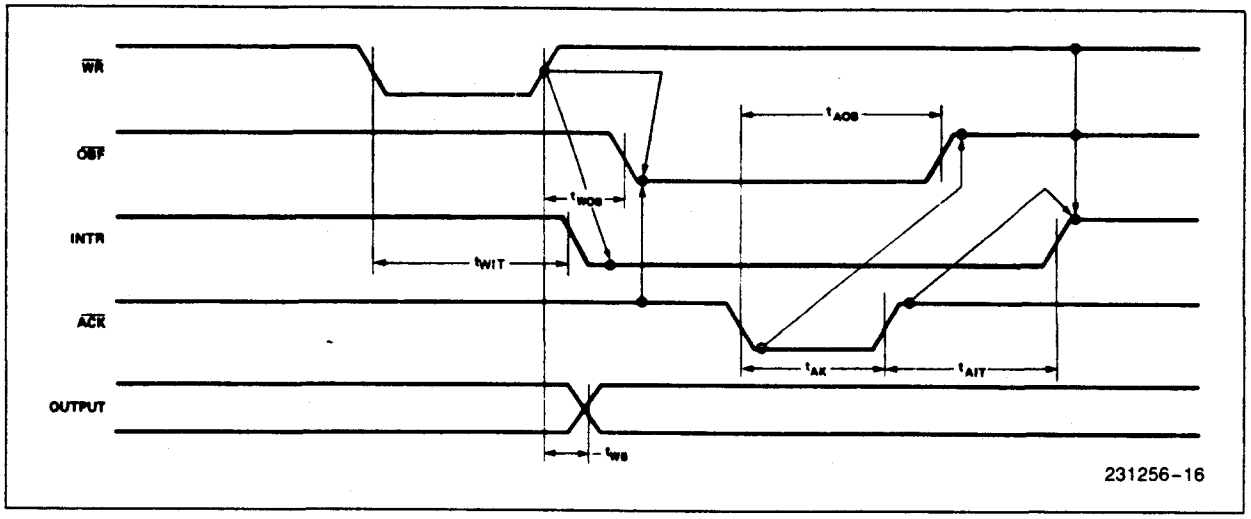


Figure 11. MODE 1 (Strobed Output)

Combinations of MODE 1

Port A and Port B can be individually defined as input or output in Mode 1 to support a wide variety of strobed I/O applications.

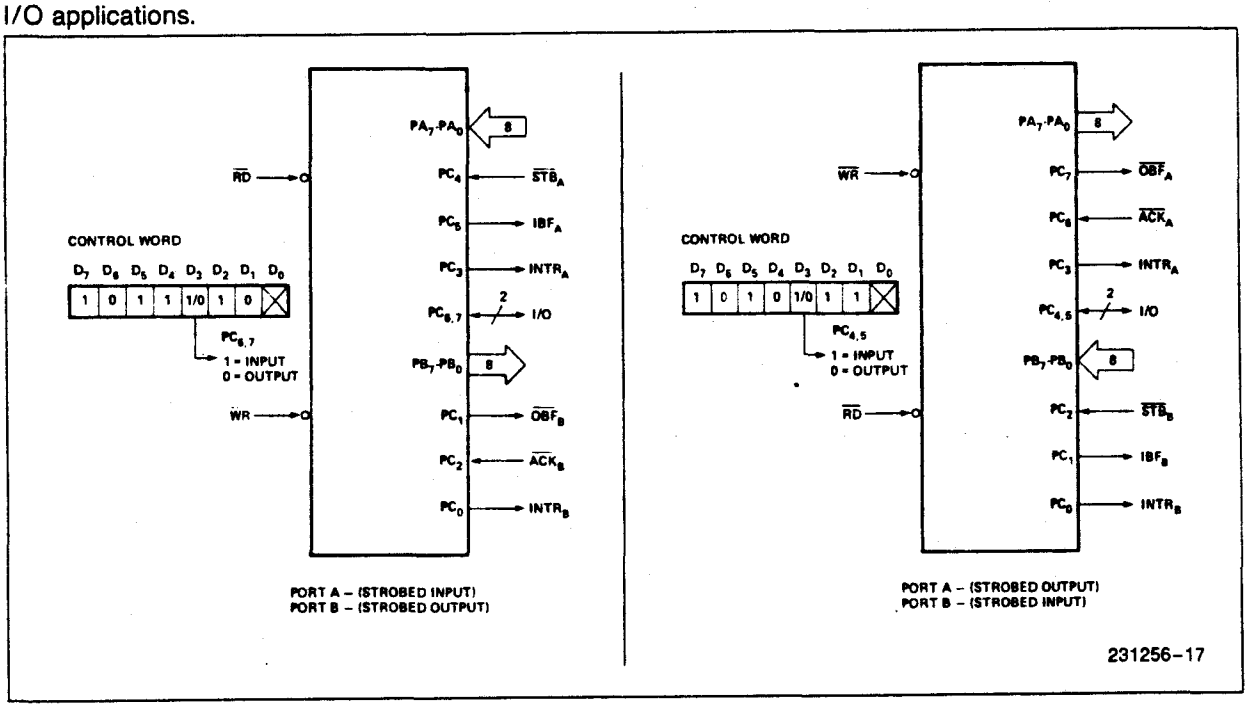


Figure 12. Combinations of MODE 1

Operating Modes

MODE 2 (Strobed Bidirectional Bus I/O). This functional configuration provides a means for communicating with a peripheral device or structure on a single 8-bit bus for both transmitting and receiving data (bidirectional bus I/O). "Handshaking" signals are provided to maintain proper bus flow discipline in a similar manner to MODE 1. Interrupt generation and enable/disable functions are also available.

MODE 2 Basic Functional Definitions:

- Used in Group A **only**.
- One 8-bit, bi-directional bus port (Port A) and a 5-bit control port (Port C).
- Both inputs and outputs are latched.
- The 5-bit control port (Port C) is used for control and status for the 8-bit, bi-directional bus port (Port A).

Bidirectional Bus I/O Control Signal Definition

INTR (Interrupt Request). A high on this output can be used to interrupt the CPU for input or output operations.

Output Operations

$\overline{OBF}$ (Output Buffer Full). The $\overline{OBF}$ output will go "low" to indicate that the CPU has written data out to port A.

$\overline{ACK}$ (Acknowledge). A "low" on this input enables the tri-state output buffer of Port A to send out the data. Otherwise, the output buffer will be in the high impedance state.

INTE 1 (The INTE Flip-Flop Associated with $\overline{OBF}$). Controlled by bit set/reset of PC_6.

Input Operations

$\overline{STB}$ (Strobe Input). A "low" on this input loads data into the input latch.

IBF (Input Buffer Full F/F). A "high" on this output indicates that data has been loaded into the input latch.

INTE 2 (The INTE Flip-Flop Associated with IBF). Controlled by bit set/reset of PC_4.

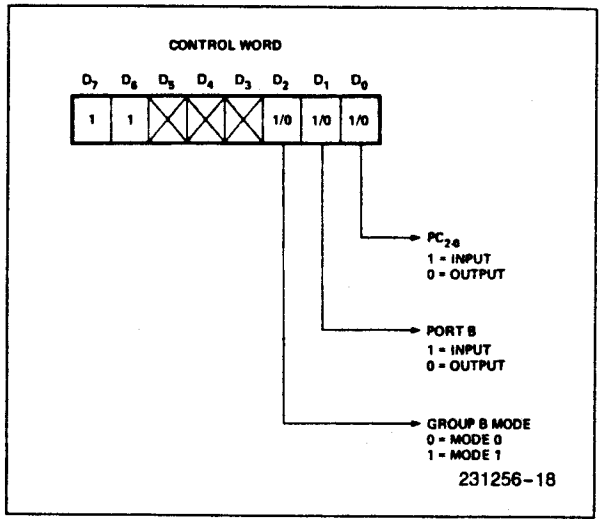


Figure 13. MODE Control Word

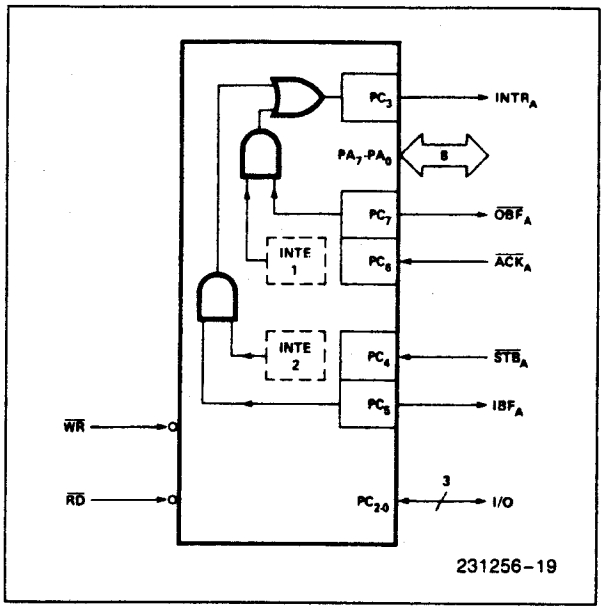


Figure 14. MODE 2

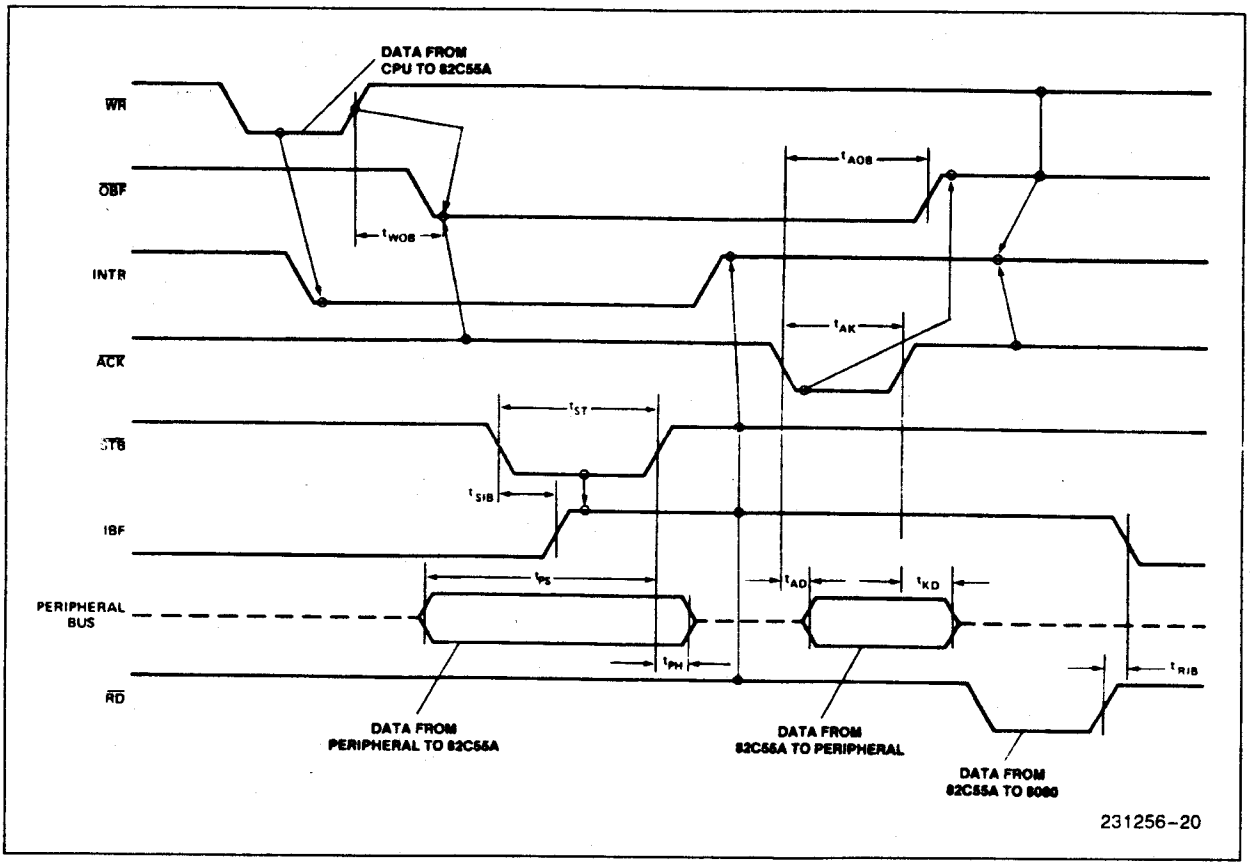


Figure 15. MODE 2 (Bidirectional)

NOTE:
Any sequence where $\overline{WR}$ occurs before $\overline{ACK}$, and $\overline{STB}$ occurs before $\overline{RD}$ is permissible.
(INTR = IBF • $\overline{MASK}$ • $\overline{STB}$ • $\overline{RD}$ + $\overline{OBF}$ • $\overline{MASK}$ • $\overline{ACK}$ • $\overline{WR}$)

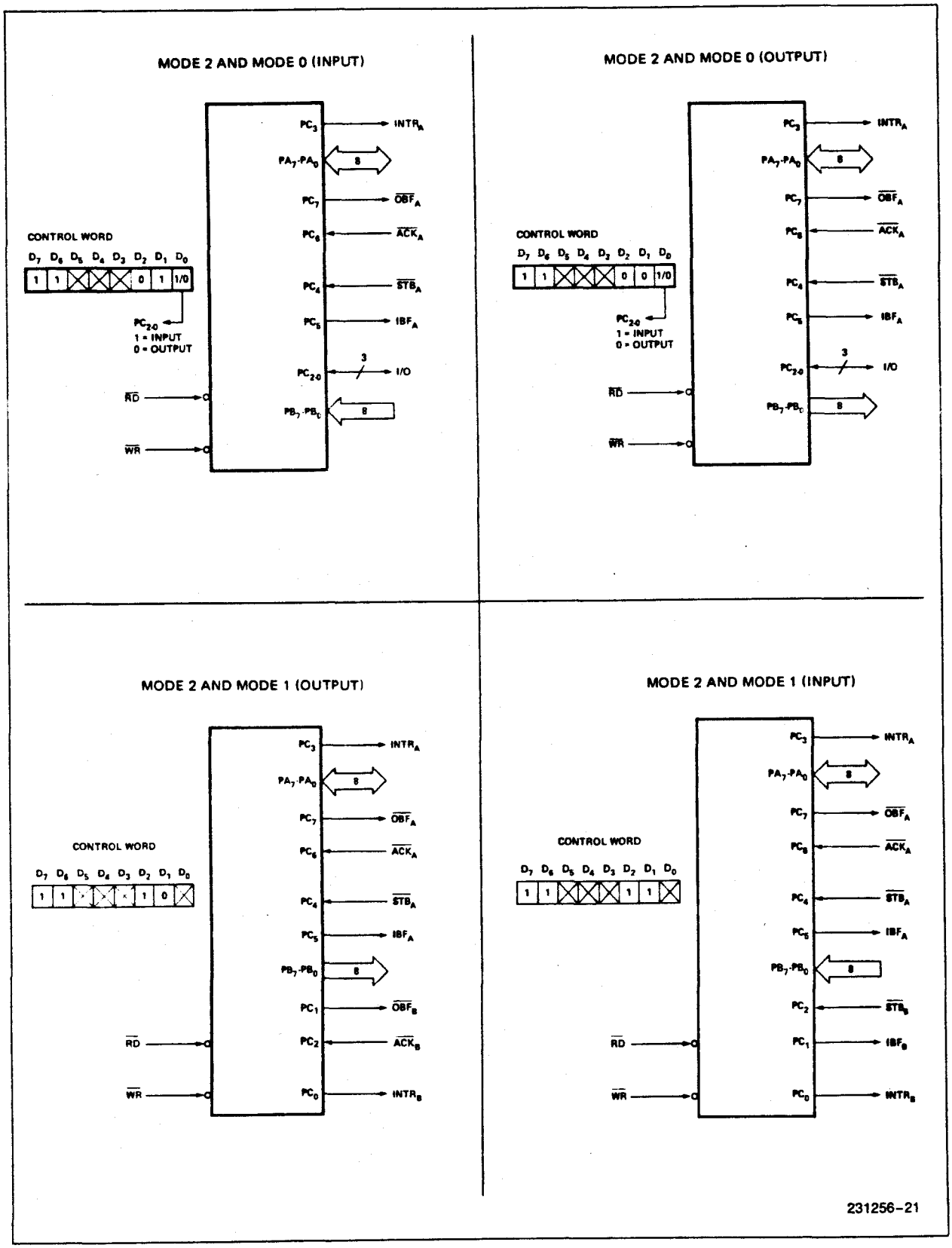


Figure 16. MODE ¼ Combinations

Mode Definition Summary

	MODE 0		MODE 1		MODE 2	
	IN	OUT	IN	OUT	GROUP A ONLY	
PA_0	IN	OUT	IN	OUT	↔	
PA_1	IN	OUT	IN	OUT	↔	
PA_2	IN	OUT	IN	OUT	↔	
PA_3	IN	OUT	IN	OUT	↔	
PA_4	IN	OUT	IN	OUT	↔	
PA_5	IN	OUT	IN	OUT	↔	
PA_6	IN	OUT	IN	OUT	↔	
PA_7	IN	OUT	IN	OUT	↔	
PB_0	IN	OUT	IN	OUT	—	MODE 0 OR MODE 1 ONLY
PB_1	IN	OUT	IN	OUT	—	
PB_2	IN	OUT	IN	OUT	—	
PB_3	IN	OUT	IN	OUT	—	
PB_4	IN	OUT	IN	OUT	—	
PB_5	IN	OUT	IN	OUT	—	
PB_6	IN	OUT	IN	OUT	—	
PB_7	IN	OUT	IN	OUT	—	
PC_0	IN	OUT	$INTR_B$	$INTR_B$	I/O	
PC_1	IN	OUT	IBF_B	$\overline{OBF}_B$	I/O	
PC_2	IN	OUT	$\overline{STB}_B$	$\overline{ACK}_B$	I/O	
PC_3	IN	OUT	$INTR_A$	$INTR_A$	$INTR_A$	
PC_4	IN	OUT	$\overline{STB}_A$	I/O	$\overline{STB}_A$	
PC_5	IN	OUT	IBF_A	I/O	IBF_A	
PC_6	IN	OUT	I/O	$\overline{ACK}_A$	$\overline{ACK}_A$	
PC_7	IN	OUT	I/O	$\overline{OBF}_A$	$\overline{OBF}_A$	

Special Mode Combination Considerations

There are several combinations of modes possible. For any combination, some or all of the Port C lines are used for control or status. The remaining bits are either inputs or outputs as defined by a "Set Mode" command.

During a read of Port C, the state of all the Port C lines, except the $\overline{ACK}$ and $\overline{STB}$ lines, will be placed on the data bus. In place of the $\overline{ACK}$ and $\overline{STB}$ line states, flag status will appear on the data bus in the PC2, PC4, and PC6 bit positions as illustrated by Figure 18.

Through a "Write Port C" command, only the Port C pins programmed as outputs in a Mode 0 group can be written. No other pins can be affected by a "Write Port C" command, nor can the interrupt enable flags be accessed. To write to any Port C output programmed as an output in a Mode 1 group or to change an interrupt enable flag, the "Set/Reset Port C Bit" command must be used.

With a "Set/Reset Port C Bit" command, any Port C line programmed as an output (including INTR, IBF and $\overline{OBF}$) can be written, or an interrupt enable flag can be either set or reset. Port C lines programmed as inputs, including $\overline{ACK}$ and $\overline{STB}$ lines, associated with Port C are not affected by a "Set/Reset Port C Bit" command. Writing to the corresponding Port C bit positions of the $\overline{ACK}$ and $\overline{STB}$ lines with the "Set/Reset Port C Bit" command will affect the Group A and Group B interrupt enable flags, as illustrated in Figure 18.

Current Drive Capability

Any output on Port A, B or C can sink or source 2.5 mA. This feature allows the 82C55A to directly drive Darlington type drivers and high-voltage displays that require such sink or source current.

Reading Port C Status

In Mode 0, Port C transfers data to or from the peripheral device. When the 82C55A is programmed to function in Modes 1 or 2, Port C generates or accepts "hand-shaking" signals with the peripheral device. Reading the contents of Port C allows the programmer to test or verify the "status" of each peripheral device and change the program flow accordingly.

There is no special instruction to read the status information from Port C. A normal read operation of Port C is executed to perform this function.

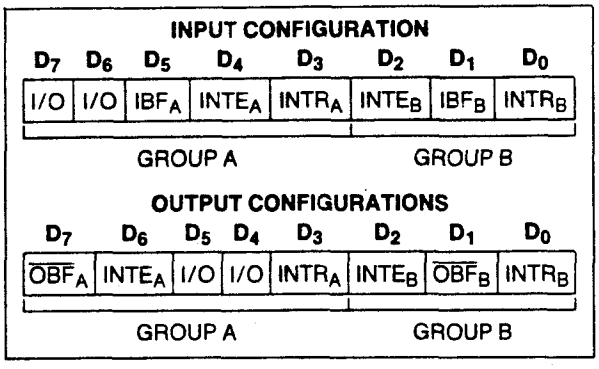

INPUT CONFIGURATION
D7 D6 D5 D4 D3 D2 D1 D0
I/O I/O IBFA INTEA INTRA INTEB IBFB INTRB
GROUP A
GROUP B
OUTPUT CONFIGURATIONS
D7 D6 D5 D4 D3 D2 D1 D0
OBFA INTEA I/O I/O INTRA INTEB OBFB INTRB
GROUP A
GROUP B

Figure 17a. MODE 1 Status Word Format

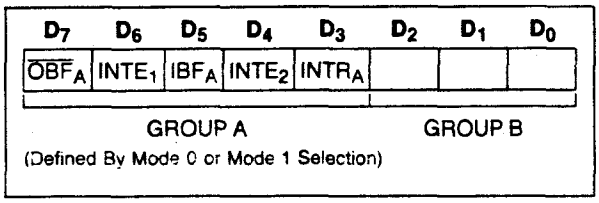


Figure 17b. MODE 2 Status Word Format

Interrupt Enable Flag	Position	Alternate Port C Pin Signal (Mode)
INTE B	PC2	$\overline{ACK}_B$ (Output Mode 1) or $\overline{STB}_B$ (Input Mode 1)
INTE A2	PC4	$\overline{STB}_A$ (Input Mode 1 or Mode 2)
INTE A1	PC6	$\overline{ACK}_A$ (Output Mode 1 or Mode 2

Figure 18. Interrupt Enable Flags in Modes 1 and 2

ABSOLUTE MAXIMUM RATINGS*

Ambient Temperature Under Bias....0°C to + 70°C
Storage Temperature .....− 65°C to + 150°C
Supply Voltage .....− 0.5 to + 8.0V
Operating Voltage .....+ 4V to + 7V
Voltage on any Input.....GND−2V to + 6.5V
Voltage on any Output ..GND−0.5V to V_{CC} + 0.5V
Power Dissipation .....1 Watt

**Notice: Stresses above those listed under "Absolute Maximum Ratings" may cause permanent damage to the device. This is a stress rating only and functional operation of the device at these or any other conditions above those indicated in the operational sections of this specification is not implied. Exposure to absolute maximum rating conditions for extended periods may affect device reliability.*

D.C. CHARACTERISTICS

T_A = 0°C to 70°C, V_{CC} = +5V ±10%, GND = 0V (T_A = −40°C to +85°C for Extended Temperture)

Symbol	Parameter	Min	Max	Units	Test Conditions
V_{IL}	Input Low Voltage	−0.5	0.8	V	
V_{IH}	Input High Voltage	2.0	V_{CC}	V	
V_{OL}	Output Low Voltage		0.4	V	I_{OL} = 2.5 mA
V_{OH}	Output High Voltage	3.0 V_{CC} − 0.4		V V	I_{OH} = −2.5 mA I_{OH} = −100 µA
I_{IL}	Input Leakage Current		±1	µA	V_{IN} = V_{CC} to 0V (Note 1)
I_{OFL}	Output Float Leakage Current		±10	µA	V_{IN} = V_{CC} to 0V (Note 2)
I_{DAR}	Darlington Drive Current	±2.5	(Note 4)	mA	Ports A, B, C R_{ext} = 500Ω V_{ext} = 1.7V
I_{PHL}	Port Hold Low Leakage Current	+50	+300	µA	V_{OUT} = 1.0V Port A only
I_{PHH}	Port Hold High Leakage Current	−50	−300	µA	V_{OUT} = 3.0V Ports A, B, C
I_{PHLO}	Port Hold Low Overdrive Current	−350		µA	V_{OUT} = 0.8V
I_{PHHO}	Port Hold High Overdrive Current	+350		µA	V_{OUT} = 3.0V
I_{CC}	V_{CC} Supply Current		10	mA	(Note 3)
I_{CCSB}	V_{CC} Supply Current-Standby		10	µA	V_{CC} = 5.5V V_{IN} = V_{CC} or GND Port Conditions If I/P = Open/High O/P = Open Only With Data Bus = High/Low $\overline{CS}$ = High Reset = Low Pure Inputs = Low/High

NOTES:
1. Pins A_1, A_0, $\overline{CS}$, $\overline{WR}$, $\overline{RD}$, Reset.
2. Data Bus; Ports B, C.
3. Outputs open.
4. Limit output current to 4.0 mA.

CAPACITANCE

T_A = 25°C, V_{CC} = GND = 0V

Symbol	Parameter	Min	Max	Units	Test Conditions
C_{IN}	Input Capacitance		10	pF	Unmeasured pins returned to GND f_c = 1 MHz(5)
$C_{I/O}$	I/O Capacitance		20	pF	

NOTE:
5. Sampled not 100% tested.

A.C. CHARACTERISTICS

T_A = 0° to 70°C, V_{CC} = +5V ±10%, GND = 0V

T_A = −40°C to +85°C for Extended Temperature

BUS PARAMETERS

READ CYCLE

Symbol	Parameter	82C55A-2		Units	Test Conditions
		Min	Max		
t_{AR}	Address Stable Before $\overline{RD}\downarrow$	0		ns	
t_{RA}	Address Hold Time After $\overline{RD}\uparrow$	0		ns	
t_{RR}	$\overline{RD}$ Pulse Width	150		ns	
t_{RD}	Data Delay from $\overline{RD}\downarrow$		120	ns	
t_{DF}	$\overline{RD}\uparrow$ to Data Floating	10	75	ns	
t_{RV}	Recovery Time between $\overline{RD}/\overline{WR}$	200		ns	

WRITE CYCLE

Symbol	Parameter	82C55A-2		Units	Test Conditions
		Min	Max		
t_{AW}	Address Stable Before $\overline{WR}\downarrow$	0		ns	
t_{WA}	Address Hold Time After $\overline{WR}\uparrow$	20		ns	Ports A & B
		20		ns	Port C
t_{WW}	$\overline{WR}$ Pulse Width	100		ns	
t_{DW}	Data Setup Time Before $\overline{WR}\uparrow$	100		ns	
t_{WD}	Data Hold Time After $\overline{WR}\uparrow$	30		ns	Ports A & B
		30		ns	Port C

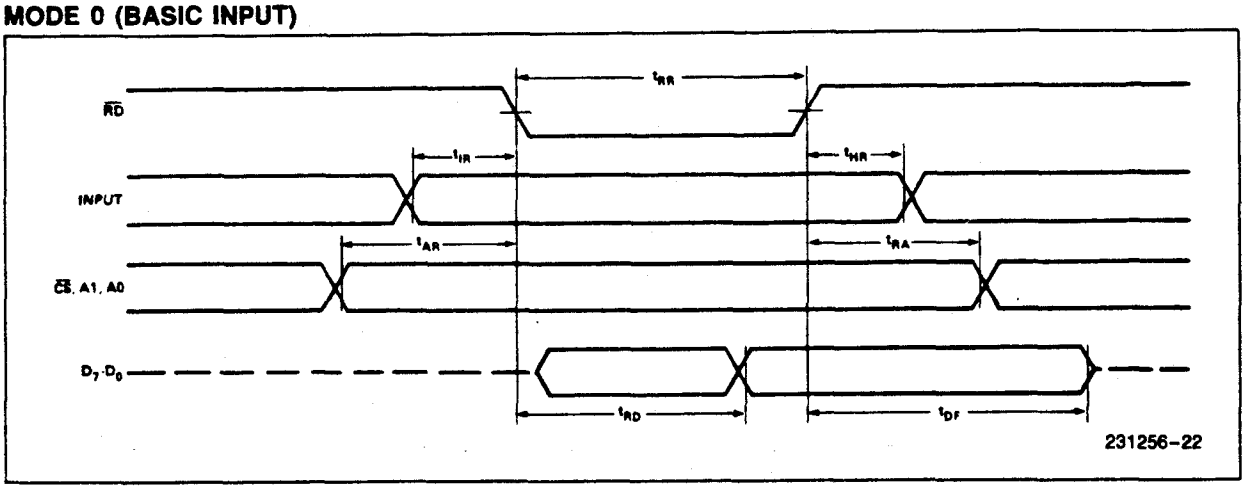
OTHER TIMINGS

Symbol	Parameter	82C55A-2		Units Conditions	Test
		Min	Max		
t_{WB}	$\overline{WR}$ = 1 to Output		350	ns	
t_{IR}	Peripheral Data Before $\overline{RD}$	0		ns	
t_{HR}	Peripheral Data After $\overline{RD}$	0		ns	
t_{AK}	$\overline{ACK}$ Pulse Width	200		ns	
t_{ST}	$\overline{STB}$ Pulse Width	100		ns	
t_{PS}	Per. Data Before $\overline{STB}$ High	20		ns	
t_{PH}	Per. Data After $\overline{STB}$High	50		ns	
t_{AD}	$\overline{ACK}$ = 0 to Output		175	ns	
t_{KD}	$\overline{ACK}$ = 1 to Output Float	20	250	ns	
t_{WOB}	$\overline{WR}$ = 1 to $\overline{OBF}$ = O		150	ns	
t_{AOB}	$\overline{ACK}$ = 0 to $\overline{OBF}$ = 1		150	ns	
t_{SIB}	$\overline{STB}$ = 0 to IBF = 1		150	ns	
t_{RIB}	$\overline{RD}$ = 1 to IBF = 0		150	ns	
t_{RIT}	$\overline{RD}$ = 0 to INTR = 0		200	ns	
t_{SIT}	$\overline{STB}$ = 1 to INTR = 1		150	ns	
t_{AIT}	$\overline{ACK}$ = 1 to INTR = 1		150	ns	
t_{WIT}	$\overline{WR}$ = 0 to INTR = 0		200	ns	see note 1
t_{RES}	Reset Pulse Width	500		ns	see note 2

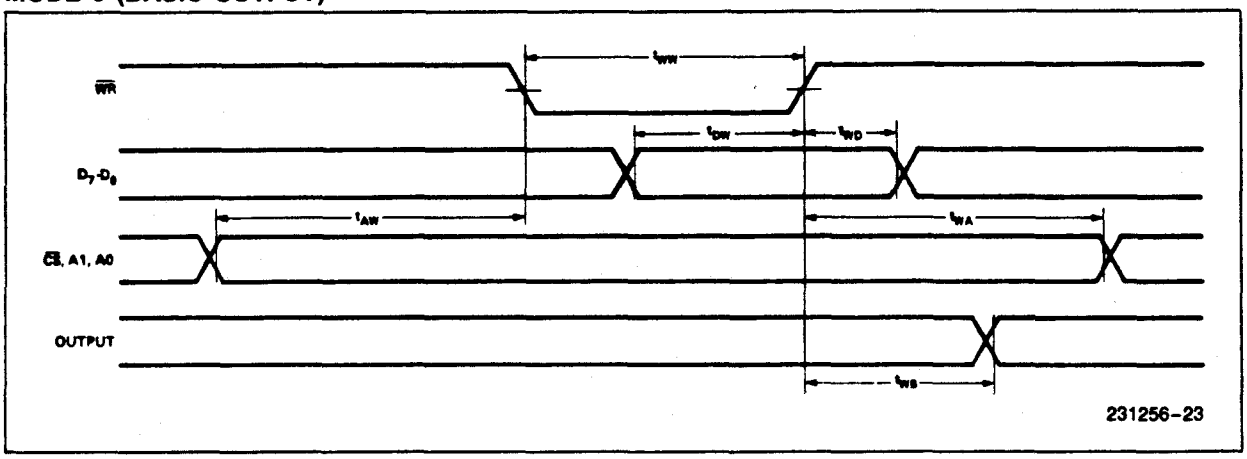
NOTE:
1. INTR ↑ may occur as early as $\overline{WR}$ ↓.
2. Pulse width of initial Reset pulse after power on must be at least 50 μSec. Subsequent Reset pulses may be 500 ns minimum.

WAVEFORMS

MODE 0 (BASIC INPUT)

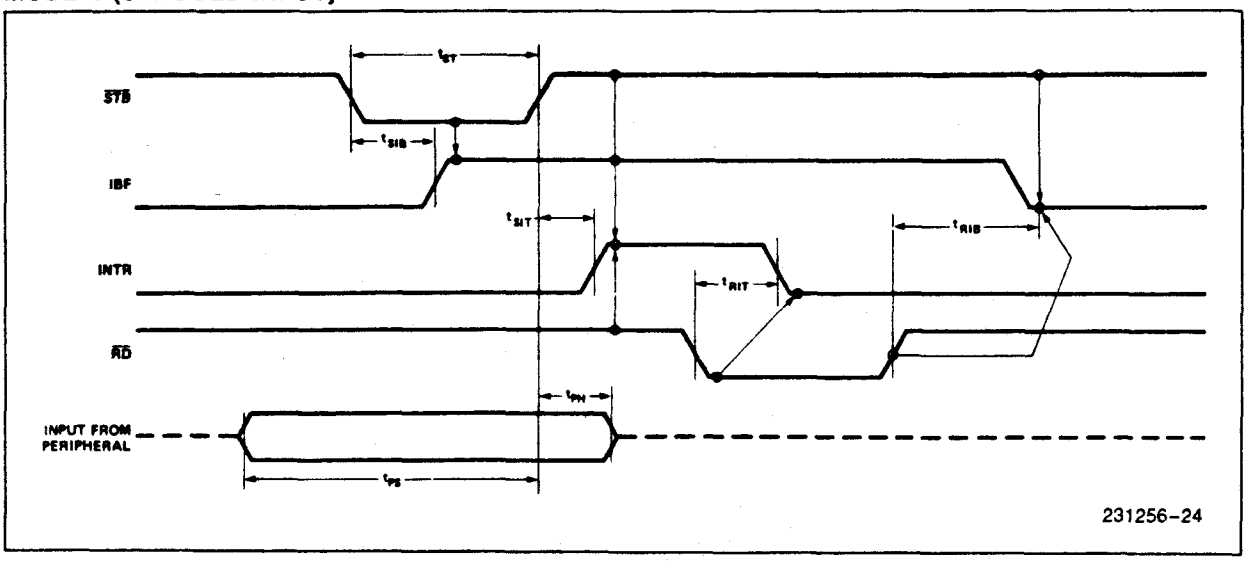


MODE 0 (BASIC OUTPUT)

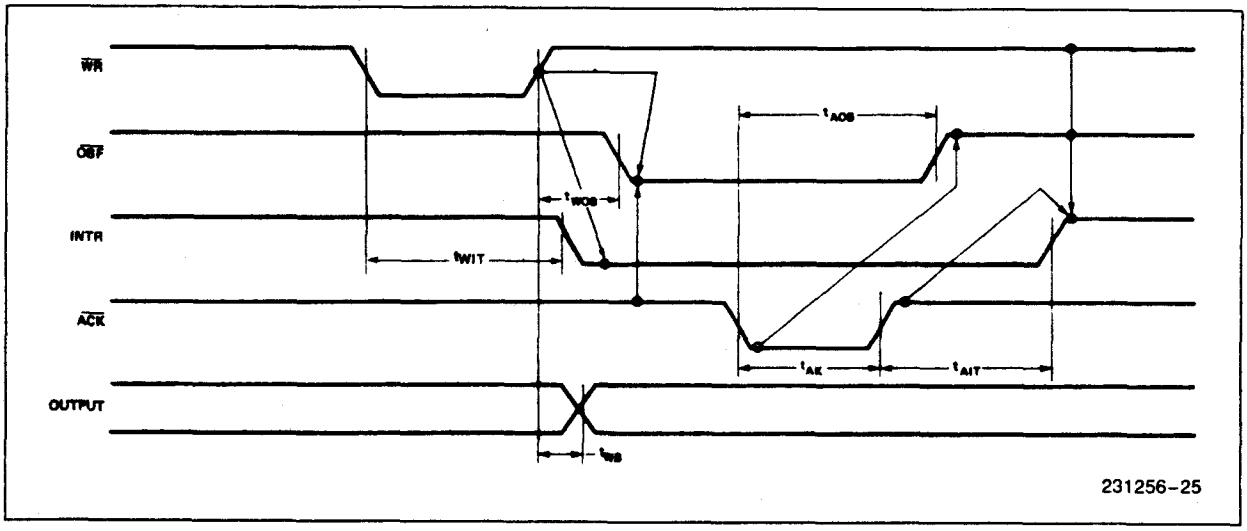


WAVEFORMS (Continued)

MODE 1 (STROBED INPUT)

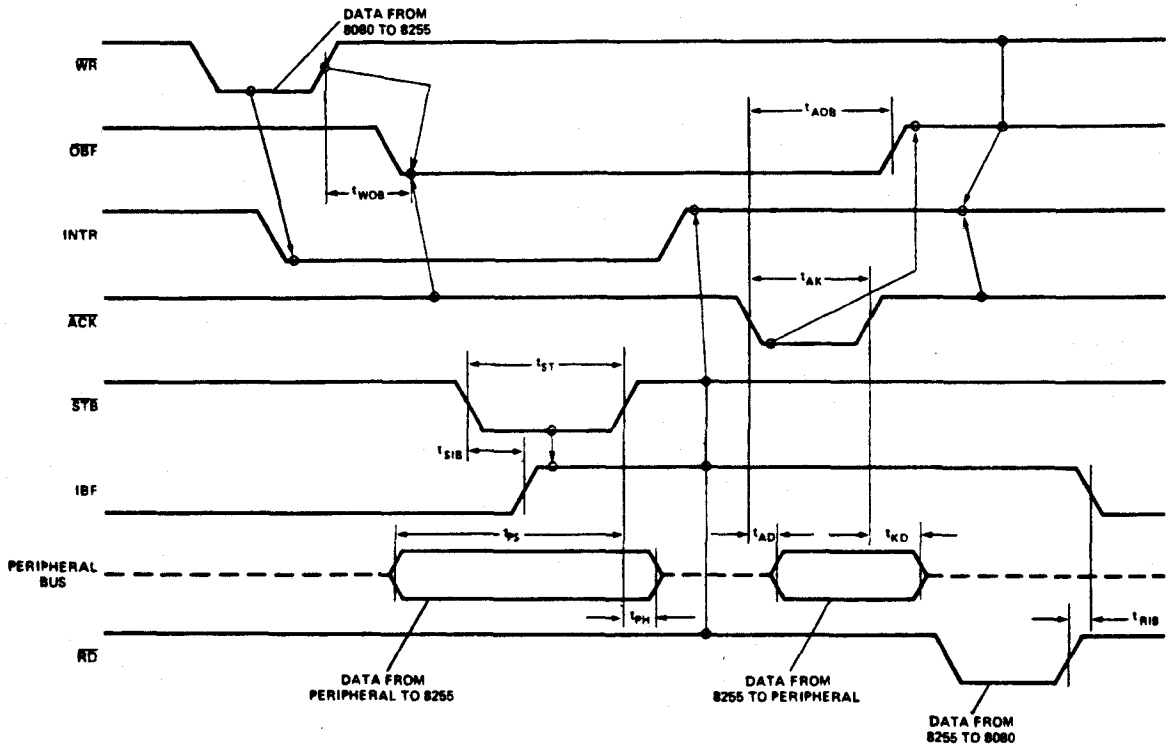


MODE 1 (STROBED OUTPUT)



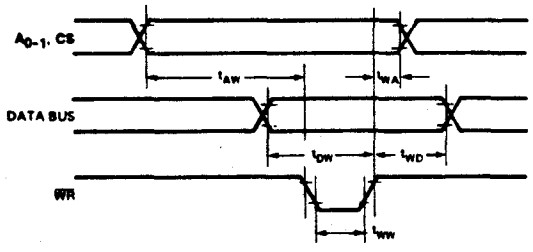
WAVEFORMS (Continued)

MODE 2 (BIDIRECTIONAL)

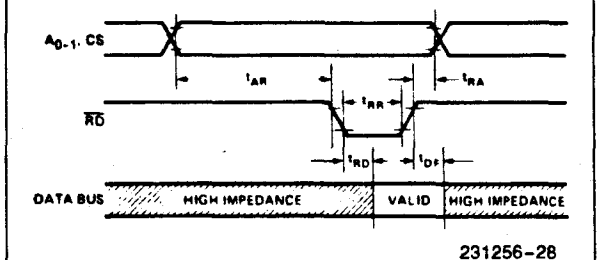


Note:
Any sequence where $\overline{WR}$ occurs before $\overline{ACK}$ AND $\overline{STB}$ occurs before $\overline{RD}$ is permissible.
(INTR = IBF • $\overline{MASK}$ • $\overline{STB}$ • $\overline{RD}$ + $\overline{OBF}$ • $\overline{MASK}$ • $\overline{ACK}$ • $\overline{WR}$)

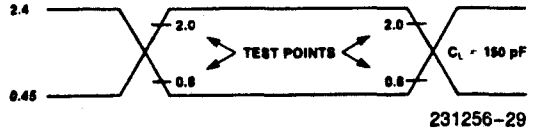
WRITE TIMING



READ TIMING

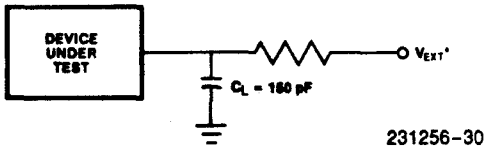


A.C. TESTING INPUT, OUTPUT WAVEFORM



A.C. Testing Inputs Are Driven At 2.4V For A Logic 1 And 0.45V For A Logic 0 Timing Measurements Are Made At 2.0V For A Logic 1 And 0.8 For A Logic 0.

A.C. TESTING LOAD CIRCUIT



*V_{EXT} Is Set At Various Voltages During Testing To Guarantee The Specification. C_L Includes Jig Capacitance.

APPENDIX D

LEXICON 510

INTRODUCTION

Lexicon 510 is an interactive program tailored specifically to make full and easy use of your AD510 analog input board. It has the characteristics of a highly specific data acquisition language and some properties of an operating system. It can be used to acquire analog data in real time and assign a time value for post-acquisition analysis. Data can be plotted or moved between the graphics templates and disk files. The files are MS-DOS compatible and can be accessed by the user's programs to perform highly specific analyses. The Lexicon 510 command set lets you characterize your experiment to determine the best acquisition rate to use, how rapidly the data changes, and how much data is being generated.

If you write your own software in BASIC, Pascal, or C, you can still use Lexicon to select a correct base address and to verify correct operation of the board.

DESCRIPTION

A Lexicon is a specialized vocabulary for a particular subject. Lexicon 510 is an extension of Forth-83, a language long used for solving complex process control and data acquisition problems. Time-critical portions of Lexicon, such as I/O, graphics, and BIOS calls are written in assembly language. Lexicon 510 is interactive, fast executing, and lets you fully control of the AD510's hardware resources so that you can become more familiar with your system. Small programs that collect data or display readings can be used to isolate and diagnose any problems such as base address contention or signal drift.

In contrast to menu-driven software, Lexicon is command-driven. Like DOS, a command typed in response to a prompt is input to the Lexicon system, interpreted, and then executed. Lexicon uses the prompt **"ok"** to let you know a command has been accepted without error. For example the command:

100 READINGS

will result in 100 A/D readings in bits being taken and displayed across the screen.

NOTE: Lexicon requires that all commands be entered in capital letters with at least one space between commands. Lexicon does not recognize lower-case letters.

The operand "100" comes before the operator "READINGS". Like Forth, Lexicon uses Reverse Polish Notation — also called RPN or postfix notation. The main reason for using a stack-oriented language is execution speed and ease of passing arguments between commands. In an RPN language, commands or programs take or leave numbers as arguments from that stack.

NOTE: On the AD510 board:

Gain	-4095 Bits	+4095 Bits
1	-5 volts	+5 volts
10	-500 millivolts	+500 millivolts
100	-50 millivolts	+50 millivolts

It is important to remember that a command like READINGS in the above example must have a 16-bit integer argument preceding it. If you type in READINGS with no argument, an error will result. How do you know if a command takes an argument on the stack? You must read the Lexicon command listed in the documentation. The Lexicon command set described later shows each command in the following format:

READINGS (n —)

On the HELP screen any command listed with a 'n' in front of it requires an argument. For example:

n READINGS

The n before the command means that a (n)umber must be entered as an argument. The valid range of numbers for Lexicon is -32767 to +32767.

LEXICON AND DOS

The Lexicon software is written to run under the MS-DOS operating system version 2.0 or greater. To insure the software will work on PC clones, much of it is written in assembly language using the BIOS and DOS software interrupts. Instead of communicating directly with the computer hardware, the BIOS ROM or DOS takes care of many common functions like drawing a pixel, reading time, changing video modes, and disk I/O.

Certain Lexicon 510 functions — such as time delays — are implemented using BIOS/DOS interrupts. For example, to tell time, Lexicon performs a BIOS call 1AH to obtain the number of 18th-second clock ticks since midnight. This means that when taking readings, the closest they can be discerned is 1/18 of a second apart. This is a limitation imposed by DOS. The correct time of day read by this interrupt is dependent on the correct time being entered when you boot your machine.

INSTALLING LEXICON 510

Before running Lexicon, we recommend that you create a separate Lexicon disk. This is done by copying the following files contained in the LEXICON subdirectory on the program disk onto a blank formatted disk:

LEX510.COM
SETUP.LEX
BLOCKS.BKS
SINE.DAT

Lexicon 510 is supplied as a .COM file and must include several other files in order to operate properly. One file that must be included is SETUP.LEX, a small file loaded after Lexicon is booted. This file contains the base address of the AD510, the default channel, and the analog input channel. SETUP.LEX can be configured for other values by the software so it can boot on an unused I/O location. BLOCKS.BKS is the Lexicon file that contains the boot instructions used by the system to set some initial characteristics and also lists the HELP screen.

Data files created by Lexicon are always followed by the extension .DAT, and have a special internal coding to insure that only Lexicon created files can be loaded by the system. SINE.DAT is an example of a data file and is included on your disk. It was generated by using an AD510 board to digitize the output of a sine wave signal generator.

To run Lexicon enter the name: LEX510 from DOS. Lexicon will load, zero the internal data buffers, load a setup file, take 15 test A/D readings, in bits, and display a HELP screen. It would be very helpful if you have a voltage source or battery connected to the input of the AD510 board. This will enable you to see if the displayed values match and track the input voltage.

LOADING THE SAMPLE DATA

To examine the contents of the disk, use the command **DIR** just as you would from DOS. To load a Lexicon file (with the extension .DAT), use:

FLOAD < name >

This command displays the directory of the active disk drive and prompts for a file name. After you enter a file name, it will be loaded into the data buffers. The command to display the data in the data buffer is:

PLOT

To exit the PLOT program, press the ESC key.

CHANGING THE DEFAULT BASE ADDRESS

If you change the base address setting of the AD510 by repositioning the jumper pin at connector P2, you must inform Lexicon that you made this change. The command to do this is **ADDR.** Lexicon must be made aware of this change or it will attempt to communicate with the AD510 at a different address and won't work at all. Lexicon boots up at base address 512 (200 hex). To change the default base address to, for example, 576 (240 hex) on boot-up, type the following:

576 ADDR
NEW

"576" is the decimal value of hex 240 on jumper P2, the second jumper setting from the left. Next type "Y" in response to the question. To perform a new initialization sequence, type the following:

LSU
SUINIT

LSU will Load SetUp and SUINIT will perform a setup initialization of the AD510. The following table shows the corresponding decimal/hexadecimal equivalents of the eight P2 base addresses:

Hexadecimal	Decimal
200	512
240	576
280	640
2C0	704
300	768
340	832
380	896
3C0	960

USING LEXICON 510

Before describing the Lexicon 510 commands in detail, there are a few items concerning the operation of Lexicon that should be repeated. All commands must be entered in upper case with at least one space between commands. The RETURN key must be pressed to execute the command(s). Lexicon is capable of handling only 16-bit integer numbers as arguments; that is, numbers ranging between -32,767 and +32,767. Numbers entered outside this range will result in indeterminate values being place on the stack.

REMEMBER:

1. Enter all commands in upper-case letters
2. All commands must be separated by at least one space

On bootup, all numbers entered and output from Lexicon are in decimal. Lexicon can also operate in hexadecimal, octal, and binary numbering modes. Internally, all numbers are stored in binary, but when entering or displaying them, you can choose a different number base using the DECIMAL, HEX, OCTAL, and BINARY commands. However, be aware that Lexicon will expect your data input to be consistent with the current number base. For example, if you are in decimal mode, the hex numbers A through F will not be accepted. Lexicon can also be used to change numbers between bases. The HEX command can be used to convert the P2 connector's hex base address values to their decimal equivalents. The following examples assume the current number base initially is decimal.

TRY THIS: HEX 2C0 DECIMAL .

DISCUSSION: The hex mode is selected and the hex value 2C0 is entered, converted to decimal and printed using the DECIMAL and . commands.

The next examples illustrate how Lexicon can change the number base of values which are input to and output from the computer.

TRY THIS: 346 HEX . DECIMAL

DISCUSSION: The number 346 (in the current number base) is placed on the stack, the number base is changed to hex, and the operator . prints the top of the stack. The command DECIMAL returns you to the decimal mode.

TRY THIS: BINARY 1101111001 DECIMAL .

DISCUSSION: The binary number base is selected and the binary number "1101111001" is entered on the stack. DECIMAL switches you back to decimal mode and . prints the decimal equivalent on the screen.

TRY THIS: DECIMAL 3147 BINARY . DECIMAL

DISCUSSION: The decimal mode is selected and the value 3147 is entered. BINARY switches the number base and . prints the binary equivalent on the screen. DECIMAL changes the mode back to base 10.

What does the operator . do? As we said earlier, . prints the top number on the stack in the current number base followed by a space on the screen.

TRY THIS: 346 .

DISCUSSION: The number 346 is entered on the stack. Regardless of the numbers already on the stack, it will go on top. Then it prints the number on the screen in the number base currently selected.

Typing . with nothing on the stack results in an error message. This is not a fatal error. Sometimes numbers can accumulate on the stack without you realizing it. You can get rid of the excess stack numbers using the . command.

LEXICON COMMAND SET

The Lexicon commands are divided into four classes: A/D Control, Operating System, File Control/Help Screen, and High-Level Commands. Letters in parentheses are the short-hand versions of the commands which can be entered in one or two keystrokes.

A/D Control. The A/D Control commands permit direct control of the AD510 board:

COMMAND	STACK	DESCRIPTION
ADC	(—)	Prints a single A/D result in bits
DIFF	(—)	Prints the DIFFERENCE between readings
?ADDR	(—)	Prints the present AD510 base address
READINGS	(n—)	Displays n A/D readings
READ (R)	(—)	Displays A/D readings continuously
IOMAP	(—)	Displays the AD510's I/O map
STATUS (ST)	(—)	Displays the current system status
ADDR	(n—)	Sets a new base I/O address
CHANNEL	(n—)	Selects the input channel number n
INIT	(—)	Initializes the AD510 board

Operating System. The Operating System commands consist of utilities to change the global system parameters such as time delay and number base:

COMMAND	STACK	DESCRIPTION
BYE	(—)	Exits Lexicon to DOS
MS	(—)	Delay according to the value in TIMER
?DELAY	(—)	Prints the present system delay
TIMER	(n—)	Select a time delay of n (dimensionless)
TX	(—)	Selects text mode on CRT
HR	(—)	Selects high resolution graphics mode
?BASE	(—)	Displays the current number base
BINARY	(—)	Selects number base of 2
DECIMAL	(—)	Selects number base of 10
OCTAL	(—)	Selects number base of 8
HEX	(—)	Selects number base of 16
DIR	(—)	Displays directory of selected drive
CLS	(—)	Clears the text screen

File Control/Help Screen. The File Control/Help Screen commands are used to save and load the extended memory array of acquired data and the SETUP array. These DOS files can also be loaded by other MS-DOS programs. A utility program (SEE) permits examination on the screen of the extended array data:

COMMAND	STACK	DESCRIPTION
FLOAD (L)	(—)	Loads a Lexicon data file
FSAVE (S)	(—)	Saves a Lexicon data file
SEE	(n—)	Examines data array starting a # n
NEW	(—)	Write present setup to SETUP file
LSU	(—)	Loads the configuration setup file
HELP (H)	(—)	Displays the Lexicon help screen

High-Level Commands. The High-Level Commands are used to collect and plot data in different formats:

COMMAND	STACK	DESCRIPTION
COLLECT (C)	(—)	Data collection program
RC	(—)	Resume data collection
ZERO	(—)	Zeroes data buffers
CHART	(—)	Strip chart simulator
FS	(—)	Full-scale plotting program
PLOT (P)	(—)	Plots data in buffers
MVS (M)	(—)	Continuous millivolt display
SCAN	(—)	8-channel scanning
A:	(—)	Selects drive A
B:	(—)	Selects drive B
C:	(—)	Selects drive C

CONTROLLING THE 510 BOARD

A useful command is READ. This command will take A/D conversions on a single channel and will continuously display them in bits on the screen. This command also has a shorthand form of 'R'. Before using READ, you must select the appropriate analog input channel. For example, **1 CHANNEL READ** will continuously display A/D readings from channel 1, and **2 CHANNEL READ** will display readings from channel 2.

Entering the command READ will cause A/D conversions to be displayed as bits. Pressing any key will halt the acquisition. Change the input voltage to the AD510 board and watch the displayed bits values change accordingly. The input range of the AD510 is -5 to +5 volts, so -5 will be -4095 and +5 will be +4096 bits for a gain of 1.

To determine the default base address, channel, or input gain, enter the command STATUS (ST).

TRY THIS: **1 CHANNEL 10 READINGS**

DISCUSSION: 1 CHANNEL selects channel 1 (AIN1) as the input channel. 10 READINGS displays 10 A/D conversions on the screen.

TRY THIS: **READ (R)**

DISCUSSION: READ (unlike READINGS) continuously performs A/D conversions and displays them in bits until any key is pressed. It uses the present gain and channel values and does not change them. Please note that since READ is often used, it has the shorthand form of "R". Type in "R" and the same command is executed.

TRY THIS: **MVS**

DISCUSSION: MVS, like the command READ, takes continuous A/D conversions and displays them in millivolts. MVS works only for a gain of 1. Pressing any key halts the conversions.

TRY THIS: **5 CHANNEL ?CHANNEL**

DISCUSSION: ?CHANNEL prints the present active analog input channel. 5 CHANNEL will set the input channel to 5 and ?CHANNEL will verify that it has been set.

TRY THIS: **DIFF**

DISCUSSION: DIFF is a program that continuously prints out the difference between two successive readings. If the analog input signal is perfectly stable, all the readings will be zero. If the signal is noisy, has base address contention, or is rapidly changing, then non-zero values will appear. This is a good debugging utility.

CHANGING THE DEFAULT VALUE

As we stated earlier, when Lexicon is loaded, a small file called SETUP.LEX is loaded and used to initialize the AD510's base address, channel selection, and gain. This file can be changed to initialize the board to your own specifications with the command NEW. NEW displays the current I/O address, active channel, and gain, and asks if you would like to make these values the setup defaults. If you respond with "Y", they are written to the SETUP.LEX file on the disk.

TRY THIS: **NEW**

DISCUSSION: NEW takes the present values of the I/O base address, channel, and gain and writes them to the file SETUP.LEX which is loaded when the system boots.

TRY THIS: **LSU SUINIT**

DISCUSSION: LSU loads the setup file from the disk into the initialization array. SUINIT is the setup initialization procedure which reconfigures the AD510 for your new setup values.

LEXICON 510 DATA FILES

Lexicon 510 saves data in DOS files. It also has the ability to create and list the files on a disk. All Lexicon created files have the extension .DAT and an internal code that identifies the file as one created by Lexicon. Files without the extension .DAT and without the internal code will not be loaded. This special encoding consists of the extension .DAT and a decimal 19 in the third byte of the file.

TRY THIS: **DIR *.DAT**

DISCUSSION: Like the DOS equivalent, DIR can accept arguments from the DOS command line. This example instructs Lexicon to display all files with the extension .DAT.

TRY THIS: **FLOAD SINE.DAT**

DISCUSSION: The command FLOAD (File Load) will load a Lexicon file into an extended array so the data can be plotted and examined.

TRY THIS: **FSAVE < name >**

DISCUSSION: The command FSAVE (File Save) will save the data stored in the extended array into the specified file (appended with the extension .DAT). It will prompt you for a file name to use. The first element of the created file is the number of data points it contains, the second is the number 19 (to code for a valid Lexicon data file). The next elements contain the system gain and delay. The acquired data follows. This command is abbreviated as "S".

Data is stored in the extended array in a special format: A/D result/time MSW (Most Significant Word)/time LSW (Least Significant Word). Every A/D conversion is tagged with a 32-bit time value (from the internal MS-DOS timer). A 64K data file will consist of about 22,000 data points and 22,000 double precision time values. The 32-bit time value represents the number of 1/18 second MS-DOS ticks since midnight.

TRY THIS: **100 SEE**

DISCUSSION: The SEE command lets you examine a data point and its 32-bit time tag. Fifty data points are displayed in this format:

Data Point. #/Time/Value (bits)

The argument for the command is the starting value you want to examine. In this example, you will begin looking at data point 100. Note that the first data point you can use with the SEE command is an argument of "2". Pressing PgUp/PgDn scrolls the readings; ESC will terminate this command.

TRY THIS: **C: DIR**

DISCUSSION: C: will select drive C as the selected drive. DIR will list the disk directory.

LEXICON 510 PLOTTING

Lexicon includes several high-level plotting programs. Because a limited number of data points can be plotted across the screen (approximately 1140), the data points are broken up into frames. One frame can be displayed on the screen at one time and it consists of two graphics boxes. The plotting programs PLOT and FS display one frame of data. The PgUp, PgDn, and Home keys are used to display the next/previous frame of the extended memory array and return to the first frame. When used with the GRAPHICS.COM utility supplied with advanced versions of DOS, high-quality graphics outputs can be dumped to a compatible printer.

TRY THIS: **PLOT**

DISCUSSION: Any data stored in the data buffers is scaled to fit into the two graphics boxes. This program displays one frame of data (which is about 1140 points). Press the PgUp/PgDn or Home key to see the next/previous frame of data displayed as a point plot on the screen. This command can be used without first FLOADing a file, but most likely you will see nothing on the display, or perhaps some random data left over from another program. Pressing the ESC key terminates it. The numbers along the X-axis of each display are the array indices of the data points. This command is abbreviated as "P".

TRY THIS: **FS.**

DISCUSSION: FS is a Full-Screen data plotting program. It displays half a frame of data. Often, data is more easily understood when a larger plot is displayed. This routine is also attractive when data is dumped using the GRAPHICS.COM utility.

TRY THIS: **CHART**

DISCUSSION: CHART imitates a strip chart recorder and will graphically display the A/D readings as a point plot in real time. ESC terminates this program.

TRY THIS: **SCAN**

DISCUSSION: The SCAN program scans and displays all eight analog input channels in bits at the selected gain. To change the delay, use the command TIMER. To change the gain, use the commands 1X, 10X, and 100X.

LEXICON 510 UTILITY PROGRAMS

In Lexicon, all high-level programs such as COLLECT, RC, CHART, and READ can have a time delay inserted to delay between each reading. To set the time delay, use the command TIMER. For example the command **5000 TIMER** stores the dimensionless number 5000 in the variable DELAY. A delay of 5000 corresponds to a delay of about 19 seconds on a 4.77 MHz machine. It can be scaled accordingly for faster CPUs. To actually get a feel for how long the delay is, use the command MS or READ.

TRY THIS: **500 TIMER READ or 5000 TIMER MS**

DISCUSSION: A value of 500 is entered into the time delay variable. This is about a 1.9 second delay between readings. It is faster for higher clock speeds and will scale up linearly.

OPERATING SYSTEM COMMANDS

The Lexicon Operating System commands are used to interface with the entire system such as clearing the screen or changing the number base.

CLS (—)

Clears the text screen; works like the same command in BASIC.

TX (—)

Selects the video text mode. If your display appears stuck in the high-resolution graphics mode, enter TX to return it to the text mode.

HR (—)

Selects the high-resolution video mode.

?BASE (—)

Prints the current number base in decimal.

BYE (—)

Exits Lexicon and returns to DOS.

HIGH-LEVEL COMMANDS — DRIVE SELECTION

A: (—)

Selects drive A as the default drive.

B: (—)

Selects drive B as the default drive.

C: (—)

Selects drive C as the default drive.

TRY THIS: C: FSAVE

DISCUSSION: Selects drive C as the default drive for listing, loading, and storing Lexicon files.

TRY THIS: A: DIR

DISCUSSION: This command sequence selects drive A as the default drive and lists the directory.

APPENDIX E

WARRANTY

LIMITED WARRANTY

Real Time Devices, Inc. warrants the hardware and software products it manufactures and produces to be free from defects in materials and workmanship for one year following the date of shipment from REAL TIME DEVICES. This warranty is limited to the original purchaser of product and is not transferable.

During the one year warranty period, REAL TIME DEVICES will repair or replace, at its option, any defective products or parts at no additional charge, provided that the product is returned, shipping prepaid, to REAL TIME DEVICES. All replaced parts and products become the property of REAL TIME DEVICES. **Before returning any product for repair, customers are required to contact the factory for an RMA number.**

THIS LIMITED WARRANTY DOES NOT EXTEND TO ANY PRODUCTS WHICH HAVE BEEN DAMAGED AS A RESULT OF ACCIDENT, MISUSE, ABUSE (such as: use of incorrect input voltages, improper or insufficient ventilation, failure to follow the operating instructions that are provided by REAL TIME DEVICES, "acts of God" or other contingencies beyond the control of REAL TIME DEVICES), OR AS A RESULT OF SERVICE OR MODIFICATION BY ANYONE OTHER THAN REAL TIME DEVICES. EXCEPT AS EXPRESSLY SET FORTH ABOVE, NO OTHER WARRANTIES ARE EXPRESSED OR IMPLIED, INCLUDING, BUT NOT LIMITED TO, ANY IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE, AND REAL TIME DEVICES EXPRESSLY DISCLAIMS ALL WARRANTIES NOT STATED HEREIN. ALL IMPLIED WARRANTIES, INCLUDING IMPLIED WARRANTIES FOR MECHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE, ARE LIMITED TO THE DURATION OF THIS WARRANTY. IN THE EVENT THE PRODUCT IS NOT FREE FROM DEFECTS AS WARRANTED ABOVE, THE PURCHASER'S SOLE REMEDY SHALL BE REPAIR OR REPLACEMENT AS PROVIDED ABOVE. UNDER NO CIRCUMSTANCES WILL REAL TIME DEVICES BE LIABLE TO THE PURCHASER OR ANY USER FOR ANY DAMAGES, INCLUDING ANY INCIDENTAL OR CONSEQUENTIAL DAMAGES, EXPENSES, LOST PROFITS, LOST SAVINGS, OR OTHER DAMAGES ARISING OUT OF THE USE OR INABILITY TO USE THE PRODUCT.

SOME STATES DO NOT ALLOW THE EXCLUSION OR LIMITATION OF INCIDENTAL OR CONSEQUENTIAL DAMAGES FOR CONSUMER PRODUCTS, AND SOME STATES DO NOT ALLOW LIMITATIONS ON HOW LONG AN IMPLIED WARRANTY LASTS, SO THE ABOVE LIMITATIONS OR EXCLUSIONS MAY NOT APPLY TO YOU.

THIS WARRANTY GIVES YOU SPECIFIC LEGAL RIGHTS, AND YOU MAY ALSO HAVE OTHER RIGHTS WHICH VARY FROM STATE TO STATE.

INDEX

A

A/D Conversion 3-3
  Conversion Modes 4-5
    Continuous Convert 4-7
    Single Convert 4-5
  MSB and LSB 4-10
A/D Converter 3-4
  Conversion Rate 3-4
Analog Input 3-3
  Input Range 3-3

B

Board Accessories *i*-3
Board Block Diagram 3-3
Board Initialization 2-3. *See also* Initialization
Board Layout 1-3

C

Calibration 5-3
  Full-Scale Adjustment 5-5
  Offset Adjustment 5-3
Cascading Boards 2-4, 4-7
Channel Selection 4-4, 4-11
Customer Service *i*-3

D

Digital I/O 2-4, 4-7

E

End-of-Convert 3-4, 4-3
Example Programs 4-13

F

Factory Settings 1-3
  P2, Base Address 1-4
  P3, Interrupt Channel Selection 1-4
  P5, External Trigger In 1-5
  S1, A/D Conversion Rate 1-5

G

Gain 1-6, 4-4. *See also* Programmable Gain
  Setting the Gain 4-4
Gain Settings 1-6

I

I/O Connections 2-3
  Analog Input 2-4
  Digital I/O 2-4
  Reset Drv 2-4
  Status 2-4
  Trigger In and Trigger Out 2-4
I/O Map 4-10
  Definition 4-10
Initialization 4-3
Installation 2-3
Interrupts 3-4, 4-7. *See also* End-of-Convert

L

Lexicon 510 D-3

O

Overvoltage Protection 3-3

P

P4 I/O Connector Pinout 2-3, B-3
Programmable Gain 1-6, 3-3
Programmable Peripheral Interface 3-4, 4-3, 4-7, C-3
  Bit Set/Reset 4-4
  PPI Control Word 4-12

S

Simultaneous Sampling 2-4. *See also* Cascading Boards
Specifications A-3
Status 4-13
Status Line 4-3

T

Technical Support *i*-4

W

Warranty E-3

AD510 Board User-Selected Settings	
Base I/O Address:	
(hex)	(decimal)
IRQ Channel Selected:	
IRQ Channel #	